

SINCLAIR

QL

WORLD

December 1992 Price £1.95

**REVIEW -
SUPERBASIC
REPAIR KIT!**

**THE
NEW
USER
GUIDE**

**SO YOU
WANT
TO BE A
SOFTWARE
AUTHOR?**

**REVIEW -
QPACER**

**REPORT -
ALL FORMATS
FAIR**

Public domain software
**WIN UPTO
30 DISKS**

ISSN 0951-9335

12 >



9 770951 933016

DJC

Dilwyn Jones Computing

*41 Bro Emrys, Tal-y-Bont,
Bangor, Gwynedd LL57 3YT U.K.
Tel: Bangor (0248) 354023*

QL SOFTWARE

NB PLEASE NOTE THE MEANING OF CODES IN SQUARE BRACKETS BELOW:

(R) RAMDISK REQUIRED
(F) AVAILABLE ON FLOPPY DISK
(M) AVAILABLE ON MICRODRIVE
(128K/512K) MINIMUM MEMORY REQUIRED.
(PC) FOR IBM PC AND COMPATIBLES.
(TK2) TOOLKIT 2 REQUIRED.

CATALOGUE

THE CATALOGUE CONTAINS MORE DETAILS OF THE PROGRAMS. CALL FOR A COPY, OR ASK FOR A COPY WITH YOUR ORDER.

DEMONSTRATION VERSIONS

Demo versions are available for some of our programs. They cost £2.00 each and this cost is refundable against the full price of the full version, see lists below.

FLOPPY DISKS & LABELS

3.5" DSDD UNBRANDED DISKS £0.40
3.5" DSHD FLOPPY DISKS £0.70
MICRODRIVE CARTRIDGES £2.50
100 DISK LABELS £2.00
100 DISK LABELS ON PRINTER ROLL £2.50
100 ADDRESS LABELS ON ROLL £2.00
100 MICRODRIVE LABELS ON ROLL £2.00
Please add £0.50 postage per roll of labels (max postage £2.50 if only ordering labels. Add £2.50 per order for floppy disks. No extra postage on microdrive cartridges.

FILE TRANSFER

DISCOVER £20.00
(F 256k) Copy files QDOS to PC disks and vice versa.
MULTI DISCOVER £30.00
(F 256k Enhanced versions, extra formats.
TEXTIDY
(F M 256k) Strip Quill files etc down to plain text.
OPD INTERCHANGE £15.00
ICL/OPD file transfer utility. Unsuitable Gold Card.
QL-PC FILESERVER £24.50
(F 128k) QL/PC software for serial cable link, access PC's drives, printer, transfer files, etc

LEISURE

SOLITAIRE £15.00
(F M 128k) A very addictive patience card game.
CRICKET SECRETARY £12.00
(F M 128k) For cricket buffs everywhere!
QUESTION MASTER £10.00
(F M 128k) For revision or entertainment.
QUIZ SETS each £5.00
FLEET TACTICAL COMMAND II QL £49.95
(F 512k) Naval strategy game, play on 2 networked QLs.
FLEET TACTICAL COMMAND II PC £69.95
(PC) Licenced PC version of this superb game.
COMBINED QL & PC VERSIONS £85.00
Package deal for both QL and PC versions together.
FTC2 DATA PRINT UTILITY £9.95
(F 128k) Print out data from FTCII described above.
THE FUGITIVE £9.95
(F 512k) Text adventure game.
COCKTAILS WAITER £10.00
(F 256k) Database of 100's alcoholic drinks recipes.
RECIPE SETS each £5.00

PROGRAMMING

S EDIT £20.00
(F 384k) DEMO VERSION £2.00
EASYPTR II £49.00
(F 256k) Pointer environment programming aid.
EASYPTR II BUDGET VERSION £35.00

DISA DISASSEMBLER

(F 256k) Interactive pointer driven disassembler. £29.00
BASIC REPORTER £10.00
(F M 128k) BASIC programming aid, list variables, etc.
BUDGET QLBERATOR COMPILER £25.00
(F M 128k) BASIC compiler for unexpanded QL.
QLBERATOR BASIC COMPILER £50.00
(F 256k) Excellent BASIC compiler for the QL.
QLOAD & QREF UTILITY £15.00
(F M 128k 128k) Fast loader and cross reference utility
MEGA TOOLKIT on disk £25.00
on EPROM cartridge and disk £40.00
Comprehensive toolkit of over 200 BASIC extensions

FILE HANDLING

LOCKSMITH £14.95
(M 128k) Microdrive backup program, byte for byte!
4MATTER £23.50
(F M 384k) Complete software backup system.
TOOLCHEST £14.95
(M F 256k) Create customised microdrive doctor
FILES 2 £12.00
(F M 128k) Simple to use popup file handling utility.
FILEMASTER £12.00
(F M 512k R) For bulk copying of files, & disk labelling.
THE GOPHER £12.00
(F M 128k) File finder, for disk, mov, or hard disk.
(F M 128k) DEMO VERSION £2.00
WINBACK £25.00
(F 256k) Hard disk and ED disk backup utility
(F 256k) DEMO VERSION £2.00

DISPLAY SOFTWARE

BANTER £25.00
(F 512k) Banner printing program for Epson etc printers.
VISION MIXER 1 £10.00
(F 512k) Screen display system.
VISION MIXER PLUS £22.50
(F 384k) Enhanced, menu driven version
PICTUREMASTER £15.00
(F 256k) Screen drawing utility for use with V.Mixer, etc.
PICTUREMASTER PLUS £20.00
(F 384k) Enhanced version, extra features.
UPGRADE TO "PM PLUS" £5.00

PRINTER DUMPS

SIDEWINDER PLUS £24.95
(F M 512k) Screen and DTP page printer dump utility.
Suitable for 9 and 24 pin printers.

GRAPHICS

THE PAINTER V4.04 £25.00
(F 512k) 100% m/code pointer driven graphics program.
THE CLIPART £12.00
(F 128k) 3 disks packed full of screen pictures.
PICTORIAL INDEX FOR THE CLIPART £4.00
New printed index for the Clipart, with screen dumps
QRACTAL £20.00
(F 512k) Machine code fractals program.
QRACTAL SCREENS £2.00
(F 128k) Sample screen pictures made with QRactal.
IMAGE PROCESSOR 2 £15.00
(F 512k) Graphics and picture editing, etc utility.
(F 512k) DEMO VERSION £2.00
PD2 CLIPART £10.00
(F 128k) 2 disks of clipart screens.
SCREEN SNATCHER £10.00
(F M 128k) Grab screen displays from other programs.
TEXT 'N' GRAPHIX £20.00
(F 256k) Print text files containing screen dumps.
(F 256k) DEMO VERSION £2.00
TRANS24 £10.00
(F M 128k) Translate 9 pin graphics for 24 pin printer.
SCREEN COMPRESSION £10.00
(F M 128k) NEW! Save space by compressing pictures.

TEXT

QTPY 2 £29.95
(F 512k) Spelling checker, uses pointer environment.
BIBLE TEXT DISKS, PLAIN TEXT FORMAT £20.00
BIBLE TEXT DISKS, QUILL FILE FORMAT £20.00
(F 512k) Text of the King James Bible on disks.
SPELLBOUND £30.00
(F M 384k) Spelling checker, non-pointer environment.
SPELLBOUND SPECIAL EDITION £50.00
(F 512k) Improved version with larger dictionary, etc
UPGRADE TO SPECIAL EDITION £30.00
Please return master disk/cartridge when upgrading.
QUICK POSTERS £10.00
(F M 128k) Text poster maker for use with Star printers.

DATABASES

ADDRESS BOOK & LABEL PRINTER. £15.00
(F M R 384k) As used by DJC, database + label printer.
(F M R 384k) DEMO VERSION £2.00
DATA DESIGN £50.00
(F 512k) Programmable pointer environment database
FLASHBACK £25.00
(F M 256k) Fast database written in machine code.
FLASHBACK SPECIAL EDITION £40.00
(F 256k) Enhanced version of Flashback
QL GENEALOGIST SECOND EDITION £30.00
(F 384k) Family trees and family history program.
UPGRADE TO SECOND EDITION £12.00
Please return master disk with upgrade order.
BUDGET 128K GENEALOGIST £12.00
(F M 128k) Cut down version for unexpanded machines.
DISK INDEXER £12.00
(F M 256k) Make a database of contents of your disks.
DBEASY £15.00
(F 512k) Database front end for Archive etc.
DBPROGS £15.00
(F 512k) A collection of Archive utilities and text files

QL HARDWARE

MINI PROCESS CONTROLLER £59.95
Relay switched outputs, controlled via QL serial ports.
SOFTWARE TOOLKIT FOR MPC £9.95
(F 128k) Extensions to simplify control the MPC.
NETWORK PROVER £3.50
A small box which plugs between networked QLs.
QPOWER REGULATOR £24.95

OTHER QL SOFTWARE

QPAC1 £19.95
(F 512k) Pointer environment utilities.
(F 512k) Files maintenance menu, hotkeys, etc.
QTOP £29.50
(F 256k) Desktop system, files, job control, etc.
PRINTERMASTER £20.00
(F M 128k) Printer control utility.
HOME BUDGET £20.00
(F M 128k) Domestic bills and accounts program.
REMIND-ME £12.00
(F M 128k) Dates reminder program. Very easy to use.
REMIND-ME PLUS £20.00
(F M 128k) Enhanced version.
SCREEN ECONOMISER £10.00
(F M 128k) Screen saver blanker.
SCREEN DAZZLER £15.00
(F M 384k) Like screen economiser, but instead of blanking out the screen activates moving objects or screen effects to save screen burn-ins, like some PC programs!
SLOWGOLD £5.00
(F 128k) Slowdown routine for Gold Card or any QL
TASKMASTER £25.00
(F M 384k) Task switching utility.
DISK LABELLER £10.00
(F 256k) Prints neat labels for your floppy disks
THE CAT £5.00
(F M 128k) List filenames in columns.
ROB ROY BARGAIN PACK £10.00
(F M 128k) Reviewed in Q L World August 1991

THE SMALL PRINT! POSTAGE AND PACKING CHARGES Software is sent post-free to UK addresses. To other countries, please add £1.00 per program for postage and packing (sent by airmail where possible). PRICES All prices are shown in UK Pounds Sterling. PAYMENT: We can accept payment by cheque (in UK Pounds Sterling currency only, please) drawn on UK branch of a bank or building society, by Eurocheque with card number written on the back, Postal Order, or by these credit cards: VISA, ACCESS, MASTERCARD, EUROCARD or by CONNECT card. Please state the card type, number, expiry date, your address, and sign orders sent by post. We can also accept orders paid by credit card over the telephone. There is an answering machine for when I am unable to answer in person so that I can call you back later. Goods remain the property of DJC until paid for in full! PLEASE STAGE IF YOU REQUIRE SOFTWARE ON 3.5 OR 5.25 INCH DISKS OR MICRODRIVE CARTRIDGE

Editor

Helen Armstrong

Publisher

Mark Kasprovicz

Advertising Manager

Jim Peskett

Creative Director

John Stanley

Graphic Artist

Stevie Billington

Magazine Services

Linda Miller, Frances

Maxwell,

Pauline Wakeling

**Sinclair QL World,
 Published by Arcwind
 Ltd.**

**The Blue Barn,
 Tew Lane, Wootton,
 Woodstock,
 Oxon. OX7 1HA**

Tel: 0993 811181

Fax: 0993 811481

ISSN 026806X

If you have any comments or difficulties please write to the editor and we will do our best to deal with your problem in the magazine, though we cannot guarantee individual replies.

Back issues are available from the publisher price £2.50 UK, £2.99 Europe. Overseas rates on request.

Subscriptions from: Arcwind
 The Blue Barn, Tew Lane,
 Wootton, Woodstock, Oxon.
 OX7 1HA
 UK: £23.40
 Europe: £32.90
 Rest of World: £40.90

Reprographic Services: The
 Wace Group, London.
 Distributed by: Seymour
 Press Ltd.,
 Windsor House, 1270
 London Road, Norbury,
 London, SW16 4DH

1992 ARCWIND LTD.
 Sinclair QL World is
 published 12 times a year. All
 rights reserved.

Reproduction in whole or in
 part without written
 permission is strictly
 prohibited. We welcome
 contributions, all material
 must be supplied with a
 S.A.E.

December 1992

Contents

6

TROUBLESHOOTER

Bryan Davies goes to a Quanta meeting and has a go at the new Conqueror.

8,9

QL SCENE

More monitors from EEC ... Merz have a mailbox ... binders available now ... Italian Fair ... SJPD expand ... Qubbesoft revive Home Finance ... All Formats diary.

10

OPEN CHANNEL

Machine code for beginners, please ... Minerva and old programs ... listings and manuals ... umlauts ... Sinclair TTL monitor problem ... new page opinions!

12

REVIEW - QPACER

Henry Orlowski tests a boot-writing facility for QPac2 users.

14

REVIEW - SUPERBASIC REPAIR KIT!

Shirley Butler reports on Xref, SuperBasic Monitor and Better Basic.

18

REVIEW - BASIC CPORT

Back from limbo - the much-admired SuperBasic CPort by Chas Dillon reviewed by Shirley Butler.

20

SHOW REPORT - ALL FORMATS FAIR

Helen Armstrong nips over to the Novotel, Hammersmith to meet the traders.

21

COMPETITION

Win up to 30 disks of public domain software - the choice is yours.

23

THE NEW USER GUIDE - part 21

This month - from ON to PARTY. Party On!

27

SO YOU WANT TO BE A SOFTWARE AUTHOR?

GCH's Richard Alexander confronts aspiring authors with everything they need to know to approach a software publisher.

32

ZX EMULATORS

The author, Carlo Delhez, describes his Spectrum and ZX81 emulators.

34

ONE MAN'S SYSTEM

Simon Goodwin, the Toolkit Man, has a number of strange machines that talk to each other.

39

A QUESTION OF DOTS - part 5

Geoff Wicks notes that, no matter how many turn up, the DTP user always wants more fonts.

42

PERFECT PARAGRAPHS AND SHINING CHARACTERS - part 3

Martin Harris concludes his proportional printing project by analysing the rules for error reports.

Coming Soon

Digital Precision's Conqueror Special Edition is out and about.

PERFECTION SPECIAL EDITION

POWER

PERFECTION SPECIAL EDITION has 253 (two hundred and fifty three) direct/menu commands (not counting options in sub-menus), plus 32 special characters (like Bold on) that can be inserted 'directly' plus intelligent (and now excellently documented) macros. Comparisons with other word processors on the subject of power are hence quite unnecessary.

EASE OF USE

Independent reports, customer feedback and published reviews (of its less able but still excellent predecessor, PERFECTION) leave one in no doubt as to which word processor is friendliest – PERFECTION SPECIAL EDITION, with its intuitive, silky handling. Uniquely, it has two operating modes, with both menus (visible or invisible – they even look like Quill's) and direct commands (for when you familiarise yourself with the system). Uniquely, both modes are 're-entrant' (so you can use any menu option or direct command while you are in the middle of performing another option or command – block handling, etc, becomes a dream). Uniquely, PERFECTION SE has fully automatic memory management, grabbing and releasing RAM instantly as your document grows or shrinks – programs without this don't take full advantage of the multi-tasking abilities of the QL! Uniquely, PERFECTION SE leaves you in the driving seat, not juggling things around 'underfoot' while you are typing. Uniquely, PERFECTION SE allows up to nine different documents to be handled simultaneously from one copy of the program – each with totally independent margin, tab, justification, control panel, etc, settings. Uniquely, each document can itself have up to six environment settings, each settable or recallable instantly with a single keypress combination. Each document can have any number (up to 500,000 on GOLD CARD) of candidate blocks! Each document can have two independent windows (of any depth, of any (but same) width across) 'on to' it, even with overlapping text – that allows you to edit in one place while viewing another, to compare 'before editing' with 'after editing' (you can arrange to have one window remain 'frozen' in time), etc. Uniquely, we realise how much faster it is to type in something like CTRL/SHIFT/F5 than (say) F3 D U – both involve three keys, but as the former doesn't require the keys to be pressed in just one specific order, or to be released in any order at all (together will do), it is in practice twice as fast as the latter, where no key may be pressed until its predecessor is released. Also, sequences like CTRL/T (top) and then CTRL/G (go to next occurrence of string in set direction) can be accomplished by holding down CTRL and then tapping T and G. Uniquely, by providing eight user-definable strips, PERFECTION SE allows you to cope with printers of the future, not just the printers that now exist – you can attach the strips to any printer features. Uniquely, PERFECTION SE's status lines give full information on all relevant global settings. And the manual has an index. Also, it has all the important bits at the front.

PC CONQUEROR GOLD SPECIAL EDITION – This terrific new product for QLs with 1.5 Mb or more makes your QL system into a PC. A well-equipped PC too, with about a megabyte of expanded RAM installed, and the ability to read, write and format SD/DD/HD/ED disks (the last by making them into pseudo hard disks). Disk performance is up to 5 times faster. Other performance is up to 55% faster than standard CONQUEROR on GOLD CARD. There are many extra features too – see our ads in June - September 1992 QLW for full details.

DR-DOS V 6.0 – The latest and most capable DOS of all!

QMATHS MATHEMATICAL SYSTEM PART TWO – An excellent complement to QMATHS, with loads of 'functionality' – fractals, function evaluation, terrain plotting, masses of maths & stats, etc.

QUICKLASER – The definitive output tool from PRO PUBLISHER to HP LaserJet II (or compatible) printers. Printed output quality subjectively exceeds that from any other QL product.

TRANSFER UTILITY SPECIAL EDITION – Does everything – 16 case change options, 14 types of sorting (multiple sorts possible), auto string translations, etc.

LIGHTNING SPECIAL EDITION GOLD CARD VERSION – See June-Aug 1992 QLW for details: optimal speed from GOLD CARD, ST/QL, THOR XVI. Free upgrade from the ROM SE version (return ROM + disk) if you are ordering something else at same time: if not, £10 charge.

SUPERB PRINT QUALITY & FLEXIBILITY

Uniquely, using the aforementioned automatic link, you can output PERFECTION SE documents using over a thousand fonts (a huge variety of styles and sizes, supplied on the PUBLISHER and TOOLBOX disks) on virtually any printer – from the humblest Epson RX80, Brother M1009 or Star LC10 (which are all single font machines when used with most word processors) to top-end lasers. *You are not limited to the fonts built into the printer!!* All PERFECTION SE **bold/underlined/italics/sup_{er}/sub**, etc, settings are preserved. Proportional spacing and micro-justification are automatic, even when you mix fonts of differing widths and heights (even on the same line), vary line spacings, etc. Uniquely, you are not trapped with one type of micro-justification (ie adding all the space between words, and using the predefined widths of characters as their separation) – with our

system, you can vary (in 5% steps) the ratio of micro-spaces added between words to that added between characters (the latter in proportion to their *individual* widths). Ratios around 65%-35% – not the 100%-0% forced upon you by some other word processors – seem to give the most pleasing results. Uniquely, you are not limited to mere rectangular columns plus headers/footers – that's all the rest can do – you can output in any sequence to any number of frames (text flowing from one to the next), each of any shape – irregular polygons of up to 66 sides, circles, multi-column or part-column boxes (hundreds of types of borders, thousands of textures), doughnuts, wrap-around shapes, even re-entrant ones ('join-the-dots' type borders, even with intersecting edges) – all with micro-justification and proportional spacing! Look at the example on this page. Of course, if super-fancy effects (like wraparound windows and mixing different font widths on the same line while maintaining right justification) are not of the essence, PERFECTION SE's direct printer output is excellent with all your printer's capabilities supported.

THE FASTEST

For benchmarking, we've used a public domain version of the first book of The King James Bible, all fifty chapters of the book of Genesis. This came to **one hundred and forty pages**, well over **two thousand words** excluding headers and footers, well over **two hundred and twelve thousand characters** excluding justification ones, **fifty full chapters** and **one thousand five hundred and thirty three indexed verses!!** We didn't use a smaller file (as used to benchmark other programs) as PERFECTION SE's timings for most operations then become impossible to stopwatch (too fast!). The hardware used for all timings was GOLD CARD: speeds would be **further improved by over three times** using the ST/QL 030. Of course, LIGHTNING SE was used. File operations were to ramdisk: normal slave blocks would give identical times. All settings on **everything** were for maximum speed, except where indicated to the contrary – we do not force full speed upon you in operations like scrolling and global Search & Replace. PERFECTION SE's speed for these is switchable (at run-time and when configuring), as too great a speed may cause overshoot (with scrolling) or fatal alteration (if there is human error inputting the target or replace strings). Here are the benchmarks for this huge document:

Load 140 pages: 0.6 seconds (yes 0.6, not 6!) ☆ Import 140 pages: 0.6 seconds (yes 0.6, not 6!) ☆ Save 140 pages: 0.5 seconds (yes 0.5, not 5!) ☆ Export 140 pages: 0.5 seconds (yes 0.5, not 5!) ☆ Case-sensitive search from top for word at bottom: 0.4 seconds (yes 0.4, not 4!) ☆ The same, but case-insensitive: 0.5 seconds (yes 0.5, not 5!) ☆ Case-sensitive search backwards from bottom for word at top: 0.4 seconds (yes 0.4, not 4!) ☆ The same, but case-insensitive: 0.5 seconds (yes 0.5, not 5!) ☆ Automatic Search & Replace, in Fast (No Query) mode, of last 600 occurrences: 7.4 seconds (same length replace string); 7.7 seconds (shorter replace string); 10.5 seconds (longer replace string – longer time as we deliberately chose a high *density* of replaces to handicap PERFECTION SE into auto-managing memory – without causing any heap fragmentation, but still with only a 0.005 second overhead per replace!) ☆ Automatic Search & Replace in Slow ('Querying') mode: arbitrarily slow, typically 30 times slower – because we deliberately allow for human response time (in case you want to abort) before proceeding from one replace to the next. ☆ Scrolling 100 lines of text, up or down, by full-width screen page: 1.5 seconds ☆ Scrolling 100 lines of text on full-width screen, line by line, in slow (full) mode: 5.7 seconds (down)/5.8 seconds (up) ☆ As above, but in medium speed mode: 4 seconds ☆ The same, but in fast mode and default settings: 13.5 seconds to scroll through the whole massive document, averaging 0.23 seconds per 100 pages (!) – and this could be made up to ten times faster by reconfiguring PERFECTION SE ☆ Reformating paragraphs, changing margins, justification, etc, of existing text: c5 times faster than predecessor ☆ Inserting (or undoing) emphasised, underlined, italics, superscript, subscript, 8 strips, 6 environment settings: Instant (i.e. immeasurable) ☆ Navigation to line or page or to top or bottom or to 8 markers or to highlights/blocks: Instant ☆ Setting new margins, justification, etc: Instant ☆ Deleting block of 100 pages: 0.3 (yes, 0.3 not 3!) seconds ☆ Copying/moving block of 100 pages (not just 10!), downwards or upwards: 3.4 seconds (yes, including all the time for automatic memory management and anti-fragmentation – other programs are light-years behind) ☆ Spellcheck as you type: Ten times faster than anyone can possibly type ☆ Spellcheck all 140 pages in the document using the 350,000 word Mega Dictionary: 3.9 seconds (20 'errors' – like 'pluckt') ☆ And using our tiny dictionary (well, tiny by our standards – large by comparison with most others): 5.1 seconds (566 'errors') ☆ Time taken to create user dictionary from the results of the second spellcheck (566 errors): 0.8 seconds to extract all 'errors' from document and clean document; 1.9 seconds to create a full user dictionary therefrom and also a sorted, duplicate-free wordlist file (for browsing) ☆ Spellcheck file (ASCII or native): Even faster. ☆ Print first 10 pages to file: 3.5 seconds. ☆ Change every occurrence in 140 pages of God to God in bold underlined italics, strip 8 – 9.5 seconds! ☆ Virtually everything else: instant.

For prices, see the coupon page of our ad. For more info, read our detailed QLW ads in early 1991 for PERFECTION, plus the extra features of the SE (well, about half of them) listed in the June-August 1992 issues. You can upgrade from the standard PERFECTION (or PLUS) to the SPECIAL EDITIONs for the difference in current price, plus £10: no manuals or dictionary disks to be returned – we'll send a supplement to the manual.

PERFECTION PERFECTION PLUS

Perfection is the finest word processor available for any computer. We have received dozens of letters from happy users saying just this... and all of these letters were unsolicited. "Superb" was used most often.

Perfection manages to achieve all the sophistication of the most complex PC word processors while still using a user interface as friendly as Quill's. Perfection has a dual system of user control: menus while you are familiarising yourself with the program, and direct commands for the time when you feel ready for more adventurous things. The two systems can be used interchangeably and even simultaneously. Even more exciting – both systems are iterative. In case you don't understand what this means, let us give you an example: suppose you wished to move a block of text using the menus. You would choose Block Move (yes, it is right in the first menu) and the screen would then tell you to move your cursor to the start of the block. On most word processors you would have to navigate manually to this position: indeed, on many of them (Quill included) only a subset of the normal navigation commands would be available. On Perfection, not only can you use all the manual navigation commands (viz all 28 permutations of CTRL, ALT, SHIFT and the arrow keys) but in addition you can use direct commands like GoTo Line or Page or any of eight markers. Even more amazingly, you can use Search (either as a direct command or from the menus) even though you are already 'within' a menu option.

Perfection has about 200 commands, but the layout of menus and the choice of keys for the direct commands makes it very easy to master. Though a 100+ page manual is provided (with all the important bits right at the front), you should only need to consult it for specialised operations like macros.

Even if speed is not particularly important to you, we assure you that Perfection's lightning performance will enable you to use the word processor in sensible ways that you would not have dreamed possible before. For example, scrolling 100 pages or so is accomplished so quickly using the normal navigation commands that you do not need to bother using a menu option to do the move. Spellchecking, assuming you have Perfection Plus, is accomplished virtually instantly: to spellcheck this whole ad (all the pages) would take under 1.5 seconds... Searching (you can switch case sensitivity, as well as equivalences between tabs, soft spaces and hard spaces) is at the rate of about 100 A4 pages per second.

Moving from one word processor to another is usually very traumatic. With Perfection, this will not be the case. Not only can Perfection read in Quill .doc and .exp files directly (you do not even need to tell it they are Quill files!) but it can make direct and immediate use of your existing Quill printer driver. File re-export is also possible.

Perfection is truly WYSIWYG: this means that bold appears bold on screen, italics appear as italics, underlined as underlined, and so on. Of course, your printer may have functions we do not know about (upside down?). To deal with these, Perfection provides a number of on-screen shaded strips: these can be attached to any printer function you wish, and will not upset justification as a translate would. Of course, translates are provided as well!

A variety of statistics on the document being processed are available: some of them are on view all the time, the rest can be toggled to instantly. Not only is there a word count, but also page, line, character and special character (like Superscript Off) counts. There are also a dozen status indicators, letting you know whether you are in Insert or Overwrite mode, whether a block is defined, whether interactive spellchecking is enabled etc. Current line (from top as well as within page) and column positions and character codes are also available.

A terrific feature of Perfection is the dual screen mode. You can view one part of the document while editing another. The sizes of the two windows are themselves adjustable, both in real-time or via the configurator. We should devote more space to the configurator: however, it must suffice to say that everything that could be dynamically set within Perfection may also be preset with the configurator. The configurator can, for example, allow you to select any of 256 colours for any of a dozen parameters (like paper colour, border colour, status window ink and paper colour etc).

Perfection is fully multitasking without need for any external accessory: however, if you already use QPAC or Taskmaster or similar and are happy, you may go on doing so.

There is absolutely no way that we can prepare you for the quality 'feel' of Perfection. We have a great deal of experience using PC word processors costing many hundreds of pounds: with absolutely no exception, Perfection is far easier to use and master.

So if you thought Perfection was unattainable, you have a very pleasant surprise coming to you!

LIGHTNING SPECIAL EDITION LIGHTNING

These programs accelerate QL operation by up to 10x (2x-4x is typical) without having any adverse effect whatsoever on compatibility or anything else. Lightning SE is typically 40% faster than the standard version. This acceleration is totally independent of, and in addition to, any speed-up obtained by hardware means. So if you have Gold Card, your need for Lightning SE is just the same as if you had only an unexpanded QL – Lightning SE will accelerate both by the same ratio.

The Lightning programs achieve their acceleration by automatically paging out sections of the QL's operating system and replacing these with optimal, concise code written by us.

Lightning installation is a completely automatic and one-off: no knowledge of computing or programming is required. Once installed, Lightning can be completely forgotten about – you will soon get used to the superb speed! Knob twiddlers are catered for too.

Lightning technology is not built in to any of our other programs. Perfection users (as well as users of all other QL software) should therefore use Lightning all the time.

In summary: If you do not have Lightning, you are wrong. Buy this one FIRST OF ALL!

PROFESSIONAL PUBLISHER

The Professional In Professional Publisher refers to the quality of output from that program, and is not meant to suggest any complexity of operation. Few programs are as easy to use as this one: > 99% of users will be able to do with-

a manual. Professional Publisher is by far the best DTP program for the QL. It is fully compatible with Perfection, Editor, Quill, Eye-Q & the ASCII editors. It allows you to both create and import both text and graphics. Text can be 'poured' into boxes of any shape, size and number, automatically maintaining justification and hyphenation settings. So flowing text around graphics is a doddle.

Professional Publisher is supplied with a generous selection of fonts of various sizes, as well as clip art.

Justification is by pixel, not by character. This gives a much smoother effect.

It is pointless for us to try to list all of Professional Publisher's features – we would end up filling half the magazine! We will concentrate on just a few 'points': Professional Publisher is extremely precise, performing all its computations accurate to a small fraction of a millimetre. All its features can be preset by you using its configurator, ruling out the need for repetitive key strokes.

The program is extraordinarily versatile while remaining intuitive in its user interface. Buy it!

PROFESSIONAL PUBLISHER TOOLBOXES

Toolbox I is an excellent collection of high definition fonts, clip art and utility programs for Professional Publisher. While the fonts supplied with Professional Publisher are excellent, many users will feel the need for a wider range of typefaces and styles.

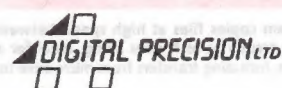
Toolbox II starts where Toolbox I leaves off, providing an even better – and different – font collection.

The two Toolboxes complement each other and are available together at a special price.

FONT ENLARGER GRAFIX

Font Enlarger does exactly what you would expect it to from its name. While Professional Publisher is also capable of enlarging fonts, it does them 'on the fly' and consequently is not able to remove the jaggedness caused by magnification. Font Enlarger is much cleverer, and enhances detail without any step effect.

While the built-in printer driver for Professional Publisher is excellent with 9-pin printers, it is not optimal with 24-pin or laser printers. Grafix is.



EYE-Q ULTRAPRINT

Eye-Q is the finest graphics program for the QL. While there may be other graphics programs with a few more features, no other program comes anywhere close to Eye-Q in sheer enjoyability. Eye-Q develops a pleasurable tactile relationship with you, and makes you feel like an artist (even if you aren't). Eye-Q graphics can be read in by Professional Publisher, and the latter's pages can be exported to Eye-Q (using Toolbox I). Everything in Eye-Q is menu-driven and there is context-sensitive help.

While Eye-Q has its own printer driver, Ultraprint allows you 22 distinct styles/sizes of printer output. The reasoning is that the scale of gradation suitable for pictures is probably unsuitable for text or line drawings.

PC CONQUEROR SOLUTION

PC Conqueror makes your QL into a PC-compatible machine, automatically. It does this by software means only, so there are no screws to undo or wires to fiddle with. Your QL stays a QL too.

Why, might you ask, should you wish to make your QL into a PC-compatible? The reason is simple: you may wish to run the same programs at home as you do at work. Alternatively, you may wish to tap into the vast storehouse of PC software of every type and description you could imagine.

Using PC Conqueror could not be easier. Just boot up your machine with the PC Conqueror disk in floppy 1 and within 10 seconds your QL will be transformed into a PC that is just waiting to be switched on. From this point on you will do exactly the same as you would if you were running a 'real' PC – this means putting a DOS disk (any version) into one of your drives and pressing a key. If you do not already have legal access to a copy of DOS, we can provide you with one at reasonable cost (see our price list).

PC Conqueror runs as fast as it is possible for a PC emulator to run: we have used all our skills to make it work quickly. Of course, you can make the emulation must faster by using Gold Card and Lightning SE. With this combination, you should get speed noticeably better than that of a PC XT...

PC Conqueror allows you to fine-tune the operating environment of the PC in order to improve performance. If you get a hard disk or other high capacity floppy system, you can utilise part or all of it as a PC hard disk.

PC Conqueror occupies under 80K and leaves 667K free for DOS when run on a Trump Card. This is more than you will get on a 'real' PC.

Solution does what Conqueror does but is about half as fast and is not quite as compatible.

SPELLCHECKER MEGA DICTIONARY

Spellchecker is what makes Perfection into Perfection Plus. We have made it available as a separate item for two reasons: (a) to allow Perfection owners to add it later (b) to allow users of other word processors to benefit from the very best in spellchecking technology.

Spellchecker is supplied complete with three dictionaries of differing sizes as well as a system for building, reviewing and maintaining user dictionaries.

Spellchecker's ultimate accessory is the Mega Dictionary, which gives the user a vocabulary of over 350,000 words!

3D PRECISION CAD SYSTEM

This program allows you to manipulate shapes and figures in 2D and 3D at a speed that will leave you breathless. Irrespective of whether your interest is in CAD, in animation or in just having fun, this program should not be missed. You can output to plotters directly from it, or alternatively create graphics screens to be manipulated and output by Eye-Q, Ultraprint or Professional Publisher.

SUPER SPRITE GENERATOR

SSG moves things about the screen very fast and very smoothly, without flicker. Sprites can have up to 16 frames.

MEDIA MANAGER SPECIAL EDITION MEDIA MANAGER

Media Manager Special Edition (MMSE) is a program to be used both when things have gone wrong as well as when things are perfectly OK. It allows for automatic, semi-automatic and manual correction of a huge variety of disk and tape problems. It allows you to explore disks and tapes to your heart's content, producing all sorts of different diagnostic reports. MMSE is very simple to operate, being menu-driven and assuming no degree of computer knowledge whatsoever.

MMSE also allows you to tidy, catalogue, sort and order your disks and cartridges.

The standard Media Manager is both less powerful and less user-friendly, but manages to work on an unexpanded QL.

Both programs allow for data transfer between PC and QL. With MMSE, this transfer is at file and directory level, is bi-directional and is completely automatic.

SPECIAL DESKTOP PUBLISHER DESKTOP PUBLISHER

These programs are quite primitive compared to Professional Publisher. However, if you have not experienced that program as yet, you will find both of these very competent. Both are capable of producing excellent results. The cheaper one has fewer features but is able to run on smaller systems.

EDITOR SPECIAL EDITION THE EDITOR

With the sole exception of Perfection, this is the best word handling system on the QL. Editor's features include an unrivalled degree of programmability and the ability to cope with the entire 256 character ASCII set. The Special Edition has enhanced document-type facilities, including column blocks and on-screen page break displays. Neither program is suitable for computing novices. Until Perfection, Editor Special Edition would have been our 'Desert Island Program'.

Editor SE can do a few things that Perfection can't, so the ideal combination is to have both (they are compatible at file level and can multitask). If you order Editor SE at the same time as Perfection, you can have Editor SE at half price.

PROFESSIONAL ASTROLOGER PROFESSIONAL ASTRONOMER

The Astrologer program teaches you Astrology from scratch and enables you to automatically produce text narrative on personality delineation, year-to-year and minute-to-minute life predictions, compatibility interpretations and so on. Whether or not you believe in astrology – indeed, especially if you do not – this program is one that you cannot afford to have. You can tailor the readouts (both in terms of quantity and what is said) to your own particular requirements. The amount of fun you can have with this program is endless. Do not blame us if you start believing in astrology, though!

Astronomer is an extremely fast and accurate solar system calculator, with planetary views, planet phases, eclipses, cinema display etc...

TURBO BASIC COMPILER

Turbo is the finest BASIC compiler for the QL and arguably the finest BASIC compiler for any computer!

Turbo automatically converts working BASIC programs into optimised machine code, usually with no need for human intervention. The benefits of this conversion are vastly enhanced running speed (as well as much faster loading, encryption and automatic bug fixing for a variety of QL Interpreter oddities). Typical speed-up is 40x – 100x.

Turbo is provided with a 200 command toolkit, adding many useful commands to BASIC. Most of these commands will be of immediate use to the programmer, whether he is a novice or an expert. There are commands to load strings and floats into RAM, and to extract them automatically; to search memory and to move its contents; to control jobs and change their priorities, manage pipes, allocate and deallocate memory, to control both rubber and virtual arrays, to present INPUT with an editable default, to have random access to files and much more.

TOOLKIT III

Toolkit III starts where Toolkit II stopped, adding about 60 new commands and enhancing many existing dual functions. Toolkit III is available either on disk or on ROM, and works whether or not you have Toolkit II.

Toolkit III commands can, with only a couple of exceptions, be compiled using Turbo.

QFLICK CARD INDEX

All QL owners have a copy of Archive, supplied free with the QL. While Archive is competent, it is very hard to get to grips with and is not particularly fast. QFlick presents a very convenient alternative – a snappy, simple-to-use, pointer-controlled card file database. You can move data between QFlick and Archive in either direction.

QFlick is not itself programmable but we document its data structure and give guidance on how to program it using Turbo.

ARCHDEV + RTM DATABASE ANALYSER ARCHIVE TUTORIAL NAMES + ADDRESSES MAILMERGE DAT-APPOINT SEdit SCREENPRINT RECOVER

This suite of utilities will greatly enhance your use of the Archive database system.

Archdev + RTM is a straight replacement for Archive: it gives enhanced speed, greater workspace and a much cleaner boot-up. All your existing applications will work.

Database Analyser provides very fast and comprehensive statistics about your Archive databases.

Archive Tutorial proceeds systematically through the whole philosophy and grammar of Archive, providing you with expert and patient guidance.

Names + addresses, Mailmerge and Dat-Appoint are ready-to-run, off-the-shelf Archive applications, providing an address database, mailmerging and appointment diary respectively. You now have no excuse not to use Archive.

SEdit allows you to create and edit screen format files in Archive. Screenprint allows you to print them out.

Recover allows you to get back lost Archive databases, created when you switched off the computer without properly exiting from Archive.

XREF SUPERBASIC MONITOR BETTERBASIC EXPERT SYSTEM

XRef analyses the structure of a BASIC program, providing detailed reports on things like variable usage, what calls what, dynamic call hierarchy of procedures and functions, and so on.

SuperBasic monitor actually monitors and reports on the performance of BASIC programs as they run under the interpreter.

BetterBasic analyses and automatically corrects structural flaws in your programs and allows you to customise things like indentation, number of statements per line, filtering out of noise words, etc.

The three programs together provide a matchless diagnostic and auto-correcting facility for BASIC programs.

TRANSFER UTILITY

This program copies files at high speed between devices, performing translates as it goes along. Ideal for all sorts of applications, including transfers from microdrive to disk.

QMATHS SYSTEM

This is an incredible mathematical compendium for the QL. Pride of place goes to the symbolic problem solver: this can solve equations, simplify expressions, factorise, expand, etc, all symbolically. If you could sneak this one into a maths examination, you would have a formidable ally. QMaths knows about all the algebraic operators, powers, roots, brackets, trigonometry, matrices, determinants, vectors, factorials, permutations, combinations, binomials, exponentials, logarithms, hyperbolics, inverse functions, infinite series including Taylor & Maclaurin expansions, complex numbers, conversions, Fourier series, and lots of calculus: both differential and integral, including integration by parts and definite integrals. QMaths optionally displays its workings and comes with a superb interactive tutorial.

The package also contains an interpretive, fractal, image-generating language with loads of beautiful fractal programs supplied for you to use and edit – no programming skill is required.

There is also a multiple precision floating point maths package, giving calculations at precisions up to over 600 decimal digits of accuracy.

There is even more to this system, but we think we have told you enough.

QMON MACHINE CODE MONITOR

The latest version of Tony Tebby's superb monitor: an absolute must for those who really want to know what is going on in the QL. No other machine code monitor even comes close.

Do not confuse this program with SuperBasic monitor, which monitors SuperBasic, not machine code.

COMPARE

This program compares files – data or program – at colossal speed. Where a mismatch is detected, the relevant areas are highlighted and you can shuffle, displace and align very easily.

CASH TRADER WITH ANALYSER PAYROLL

Cash trader with Analyser is an accounts system designed by businessmen and not by wretched accountants! Consequently, it has excellent reporting and management facilities, and is very flexible. It is aimed primarily at the layman, probably a sole trader running a small or medium sized business. All the features you would expect – including audit trail – are present.

Payroll is a reasonably flexible system designed to automate the payroll function in small businesses.

Both programs are configurable, with editable defaults letting you adapt the programs from year to year.

HARDBACK WITH FINDER

This is the ultimate hard disk backup and management utility, with all the sophisticated features you could want. User dialogue is via overlapping pop-up windows – the whole program just feels right. It is possible to scan the disk at great speed, too.

DISKTOOL WITH QUICKDISK

This permits you to add password protection to disks, to optionally increase disk storage capacity on DSDD drives by 36K and to increase speed of access by as much as 30%. All this is done while maintaining full compatibility. Automatic file management is also provided.

DIGITAL C SPECIAL EDITION DIGITAL C

These are extremely fast and efficient C compilers, complying with and surpassing the Small C definition. The Special Edition goes much further, including support for structures, pointers, long pointers, >64K code size, direct access to QDOS traps, etc. The Special Edition C generates code that runs about twice as fast as the other.

SPECIAL DEALS

5% off total if you buy 2 programs/upgrades;
10% off 3; 15% off 4; 20% off 5; 25% off 6+
Upgrades cost difference in price + £10
Non-UK Europe add 5%, rest of world 10%

For full terms and conditions, please refer to any of our QL World ads from Jan-Nov 1990, or write in including a SAE

CPORT IMPROVED VERSION

A brand new CPORT system, enabling you to rapidly convert your SuperBASIC programs into C (ANSI or Lattice). The new (October 1992) version is now as close to being fully automatic as makes no difference – you must get it!

Owners of our earlier CPORT versions should return disk + SAE for a free upgrade.

SUPERFORTH COMPILER WITH REVERSI

Forth is the most logical computer language. This compiler produces multitasking code. The manual teaches you Forth-83 from scratch.

IDIS SPECIAL EDITION IDIS

These intelligent disassemblers make the otherwise terrifyingly complex task of understanding other people's machine code programs absurdly easy. The SE version, which has a higher hardware requirement, sorts out some routines, replaces addresses with names, untangles data from code and much more.

QKICK FRONT END SYSTEM

This is a simple, easy-to-master, pull-down menu controlled multitasking front end. QKICK runs in the background and can be called up at any time. It provides you with notepads, sophisticated file/sector/RAM handling, backing up facilities, a clock, diary, calculator, mini-database and so on.

ADVENTURE CREATION TOOL SPECIAL EDITION

ACT is a must for every programmer. The name of the program is misleading, insofar as it has capabilities far beyond the 'mere' creation of adventures. ACT has utilities providing animated graphics, data compression, language design, parsing, maps, object-oriented control etc. If all you want to do is generate adventures, though, you do not need to be a programmer to use it. This is a purchase you will never regret.

PEDIT

A fast, modern and capable printer driver for the programs bundled with the QL.

MICROBRIDGE

Superb contract bridge bidder (ACOL etc) and player, using millions of random but reconstructable hands. Microbridge also includes a state of the art interactive bidding tutor and a clear instruction manual. There is nothing like this anywhere else!

SUPER ASTROLOGER

A very cut-down version of Professional Astrologer – still great fun, though!

SUCCESS CP/M EMULATOR

Allows your QL to run CP/M programs at great speed.

3-D PRECISION CAD SYSTEM	£ 49.95	d
ADVENTURE CREATION TOOL SPECIAL EDITION	49.95	e
ARCADIA GAME	9.95	f
ARCHIVE DEVELOPMENT SYS + RUN-TIME MODULE	29.95	a
ARCHIVE TUTORIAL	19.95	a
BETTER BASIC EXPERT SYSTEM	24.95	a
BLOCKLANDS GAME	9.95	f
CASH TRADER v3.3 + ANALYSER	99.95	c
COMPARE FILE UTILITY	19.95	a
COPY UTILITY FOR BACKING UP	9.95	a
CPORT BASIC TO C TRANSLATOR WITH C68K COMPILER	99.95	c
CPORT BASIC TO C TRANSLATOR	89.95	c
DATABASE ANALYSER FOR ARCHIVE	19.95	a
DAT-APPOINT APPOINTMENT DATABASE SYSTEM	19.95	d
DESKJET/LASERJET DRIVER FOR PRO PUBLISHER	19.95	c
DESKTOP PUBLISHER SPECIAL EDITION	39.95	c
DESKTOP PUBLISHER	24.95	d
DIGITAL C COMPILER	29.95	a
DIGITAL C SPECIAL EDITION	49.95	a
DISKTOOL WITH QUICKDISK	19.95	b
DROIDZONE GAME	9.95	f
DR-DOS v6.0 STATE OF THE ART PC OPERATING SYSTEM	79.95	e
EDITOR SPECIAL EDITION	49.95	d
EDITOR	29.95	a
EYE-Q GRAPHICS SYSTEM	39.95	a
FONT ENLARGER	19.95	e
GAMES COMPENDIUM (ALL FIVE GAMES)	29.95	a
GRAFIX 24-PIN UNIVERSAL PRINTER DRIVER	14.95	e
HARDBACK + FINDER HARD DISK MANAGER	49.95	c
IDIS INTELLIGENT DISASSEMBLER	24.95	a
IDIS SPECIAL EDITION	39.95	d
LIGHTNING ACCELERATOR	24.95	a
LIGHTNING SPECIAL EDITION ACCELERATOR	49.95	h
LIGHTNING SPECIAL EDITION FOR GOLD CARD	39.95	a
MAILMERGE DATABASE SYSTEM	19.95	a
MEDIA MANAGER SPECIAL EDITION	49.95	d
MEDIA MANAGER	24.95	a
MEGA DICTIONARY	29.95	g
MICROBRIDGE	39.95	d
NAMES AND ADDRESSES DATABASE SYSTEM	19.95	d
PAYROLL SYSTEM	49.95	c
PC CONQUEROR GOLD SPECIAL EDITION WITH DR-DOS v6.0	179.95	g
PC CONQUEROR GOLD SPECIAL EDITION	99.95	g
PC CONQUEROR WITH DR-DOS v6.0	139.95	e
PC CONQUEROR	59.95	e
PEDIT PRINTER DRIVER FOR XCHANGE	19.95	a
PERFECTION PLUS SPECIAL EDITION WITH SPELLCHECKER	139.95	c
PERFECTION PLUS WITH SPELLCHECKER	99.95	e
PERFECTION SPECIAL EDITION WORD PROCESSOR	99.95	c
PERFECTION WORD PROCESSOR	59.95	d
PERFECT POINTER TOOLS	29.95	a
PROFESSIONAL ASTROLOGER WITH ASTRONOMER	69.95	a
PROFESSIONAL ASTROLOGER	59.95	a
PROFESSIONAL ASTRONOMER	29.95	f
PROFESSIONAL PUBLISHER TOOLBOXES (PARTS ONE & TWO)	49.95	e
PROFESSIONAL PUBLISHER TOOLBOX PART ONE	29.95	e
PROFESSIONAL PUBLISHER TOOLBOX PART TWO	29.95	e
PROFESSIONAL PUBLISHER	89.95	c
QFLICK CARD INDEX SYSTEM	29.95	a
QKICK FRONT END SYSTEM	24.95	a
QMATHS MATHEMATICAL SYSTEM PART ONE	69.95	c
QMATHS MATHEMATICAL SYSTEM PART TWO	59.95	c
QMATHS MATHEMATICAL SYSTEM (PARTS ONE & TWO)	99.95	c
QMON MACHINE CODE MONITOR v2.05	39.95	a
RECOVER ARCHIVE UTILITY	19.95	a
REVERSI GAME	9.95	f
SEDIT + SCREENPRINT ARCHIVE UTILITIES	39.95	a
SOLUTION PC EMULATOR WITH DR-DOS v6.0	109.95	e
SOLUTION PC EMULATOR	29.95	e
SPELLCHECKER WITH DICTIONARIES	49.95	d
SUCCESS CP/M EMULATOR	49.95	b
SUPERBASIC MONITOR	24.95	a
SUPERFORTH COMPILER WITH REVERSI	39.95	a
SUPER ASTROLOGER	24.95	f
SUPER BACKGAMMON GAME	9.95	f
SUPER SPRITE GENERATOR	29.95	a
TOOLKIT III WITH ROM	49.95	h
TOOLKIT III	29.95	a
TRANSFER UTILITY SPECIAL EDITION	29.95	b
TRANSFER UTILITY	9.95	b
TURBO BASIC COMPILER + TOOLKIT	79.95	a
ULTRAPRINT SCREEN DUMP UTILITY	19.95	a
XREF SUPERBASIC PROGRAM ANALYSER	29.95	a
KEY>>>>>> Available either on cartridge or disk		a
Available only on disk		b
Minimum 512K exp:only available on disk		c
Minimum 256K exp:either cartridge or disk		d
Minimum 256K exp:only available on disk		e
Available only on cartridge		f
Minimum 1.5Mb RAM:only available on disk		g
As well as cartridge or disk, you get a ROM		h

NEW!

Send to: DIGITAL PRECISION LTD, 222 THE AVENUE, CHINGFORD, LONDON E4 9SE or phone 081-527 5493

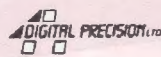
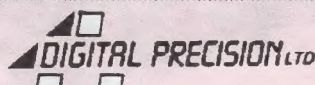
Please rush me:

Name: Address: Postcode:

Encl: CHEQUE/VISA/ACCESS/MASTERCARD/PO/MO/CASH for £

Card No: Expiry: Signed:

Delete as appropriate: (QL/THOR/STQL)(MDV/3.5"/5.25")(Disk=360/720/1440/2880/6400 sectors; nb 1 sector=0.5K)(If 720 sectors: SSDD/DSSD)



All Formats Diary

The All Formats Computer Fairs dates for the spring season have been released. The dates for January and February are:

Jan 17: West Midlands National Motorcycle Museum, M42 junction 6; Jan 23 North east Northumbria Centre, Washington; Jan 24: Northern University Sports Centre, Calverley St., Leeds; Jan 30: Nottingham Jesse Boot Centre, the University; Feb 6 London Novotel, Hammersmith (by Hammersmith underground station, various lines); Feb 7 Wales University Union, Park Place, Cardiff; Feb 13 London Sandown Park, Esher, Surrey, M25 junctions 9 and 10; Feb 14 West Midlands National Motorcycle Museum, M42 junction 6; Feb 20: Haydock Park Racecourse, M6 junction 23; Feb 21 Scotland City Hall, Candleriggs, Glasgow; 27 Hemel Hempstead Dacorum Pavilion, The Marlowes Centre; 28 Feb Western Brunel Centre, Templemeads Station, Bristol.

Check with any particular supplier that you are hoping to see whether they will be at a particular Fair. Many QL suppliers only attend the Scotland and London fairs with any regularity. The Hammersmith venue is preferred in London.

Out of the Hat?

May we quash here a rumour that Dilwyn Jones is really called "Dillon"? Nothing could be further from the truth, he says. He's related neither to the Magic Roundabout's philosophical rabbit, nor to Chas of that name. Heavens! We know that Brits have trouble pronouncing foreign names, but the boyo only comes from Wales.

On the other hand, when you put it like that ...

Stop Press - Binders!

Our office has just ordered another batch of popular QL World magazine binders, so now is a good time to indulge yourself!

Italy Fair in January

The **Fourth International QL Meeting in Italy** will take place on Sunday 24 January 1993. The organisers are Davide Santachiara of Ergon Developments at **Via Emilio de Marchi 2, 42100 Reggio Emilia, Italy (Tel. +39 522 70409)** or Eros Forenzi of **QItaly magazine at Via Valeriana 44, 23010 Berbenno Sondrio, Italy (Tel. +39 342 492323, 1900 to 2200 hours seven days a week)**. They can also be contacted on the QL areas of Fidonet, BBS matrix address 2:335/21 for fast private messages.

Reggio is well served by train from other parts of Italy. More from Ergon next month.

Merz has a Mailbox

Jochen Merz Software have introduced their own Qdos mailbox on 0049 203 5917806 (German), 24 hours with up to 9600 baud. Jochen appeals for contacts and assistance getting the service going, and points out that a quick upload/download with a support query is as cheap as a letter, and faster. Real update support will be available through the Mailbox to customers who subscribe to it from for beginning of 1993, avoiding postal delays on update disks. Apart from this, we gather, use of the BBS is free (apart from the usual phone bills, of course). An information page will give information on the latest version of all Merz programs.

QSpread is now in version 1.08 with a new easy-percentage calculation. Free updates if the master disk and two International Reply Coupons are sent. Merz is now quoting prices in DM only to avoid currency fluctuations. Convert prices at the current local rate (ask your bank or see a newspaper), not forgetting any postal charges etc.

Jochen Merz Software, Im Stillen Winkel 12, 4100 Duisberg, West Germany.

EEC Goes Again With Special QL Monitors

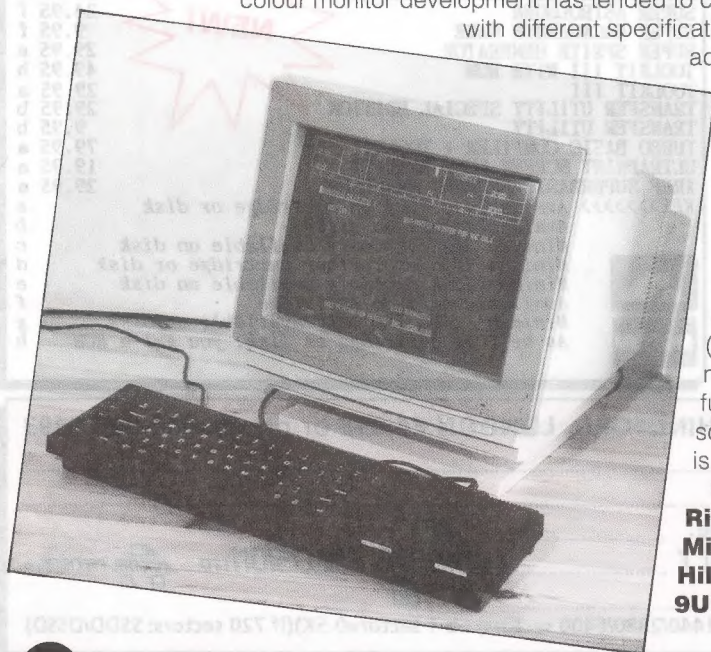
Bill Richardson of W N Richardson & Co. (EEC) is very pleased to announce that the company is one again able to supply the 14in Goldstar Colour Monitor for the QL. EEC's last batch sold out quickly because its hi-res screen and ability to show the QL's 85-column width made it a very popular buy with QL users.

The 14in Goldstar is of particular interest to users who are still using a TV or monochrome monitor for wordprocessing or spreadsheets, as it produces a dramatic improvement in viewing and resolution at the reasonable price of £199 - inclusive of VAT (carriage is £9 extra). The original QL monitors were made by Microvitec, and have been out of production for a long time. Recent colour monitor development has tended to concentrate on PC requirements,

with different specifications, which have been difficult to adapt for QL use. The Goldstar stocked by EEC has proved suitable.

The Goldstar EGA monitor can be used for a PC if readjusted, but has been modified for immediate use with the QL, and provided with a special adaptor lead to plug into the QL monitor socket. The result is enhanced (EGA) graphics quality with the necessary ability to show the QL's full 85-character width on the screen when the F1 (monitor) option is selected.

Orders and enquiries to **WN Richardson & Co.(EEC), 18-21 Misbourne House, Chiltern Hill, Chalfont St. Peter, SL9 9UE. Tel. 0753 888866.**



Qubbe takes on Home Finance

Ron Dunnett of Qubbesoft public domain library now has in stock two QL versions of an early commercial success, *QL Home Finance* by Francis Ainley. One version of the program is written for the basic 128K QL. The other version works with the Miracle Systems Trump Card. "Unfortunately", says Ron, "There is no version for the Gold Card", as the program was developed a long time before the most modern QL hardware. QL Home Finance is on Qubbesoft's Disk Special 9, costing £1 on 3.5 in disk only, or 50p if a disk is sent. Postage is extra.

Qubbesoft have also recently acquired the shareware version of the ZM1+ ZX Spectrum Emulator from Ergon Developments in Italy. We may even be able to squeeze in a snap of Qubbesoft's catalogue cover here ...

Ron also reports that the Miracle Trump Card, which he took over from Miracle a few months ago, is still selling despite the hot-cakes success of its smarter younger brother the Gold Card.

Qubbesoft are at **38 Brunwin Road, Rayne, Braintree, Essex CM7 5BU. Tel. 0376 347852.**



Motorola pursue the 68000

Motorola, makers of the 68000 series processors around which the QL is designed, have pledged that there will be at least two more processors in the 68000 family following the 68060, which should itself appear in 1994.

But the new devices, being developed at present under the code-names MP and XP, are aimed at control applications rather than personal microcomputers. Motorola's avowed intent is to go for cost-effectiveness in the industrial arena, rather than the highest performance in the field of personal computer processors.

Nevertheless the planned developments demonstrate that Motorola is continuing to support the 68000 family. Meanwhile, QL-based companies like Miracle Systems are continuing to palm and base their developments on the current and projected processor upgrades in the 68000 clan.

Archive erratum

Half a line dropped out of the Archive proc NewPage on page 30 of the June *QL World* (as Robin and various readers have recently pointed out). The line should have read:

let line=1: endif : endif

SJPD take on Ergon

SJPD public domain library, this month's competition sponsors, are also publishing the latest public domain version of the Ergon Developments ZM1+ Spectrum Emulator on their disk no. SJS28, at a copy charge of £1 plus return postage (customer's disk) or £1.75 (disk and postage inclusive).

SJPD's PD library features around 90 disks of public domain software and shareware. See page 21 for our 60-disk prize competition.

Minerva, Hermes still growing

A new version of the recently-released Minerva MkII, version 1.94, will be released soon to allow split output baud rates. Used with the Hermes intelligent peripheral controller, this will give independent baud rates on each port, both input and output. A revised and indexed Minerva manual is under preparation.

A German version of Hermes is now being distributed by Albin Hessler Software, including some commands written by Hessler. These include more straightforward commands to set input baud rate. Hermes allows the use of 14400 bps modems (V32bis and HST) directly connected to the QL's ser2 port.

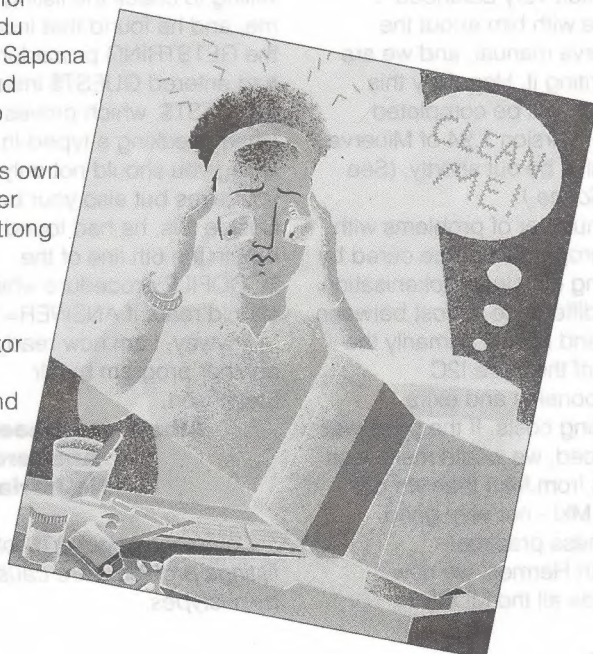
Minerva and Hermes are distributed by **TF Services, 12 Bouverie Place, London W2 1RB, UK. Tel. 071 724 9053.**

Screen alert

An office equipment supplier called Sapona are launching a new range of specialist screen cleaning products for monitors, "to help in the fight against vdu eyestrain, headaches and fatigue. The Sapona range, which includes sealed wipes and aerosol sprays, has been developed to remove grease and other dirt from vdu screens without destroying the screen's own protective coating (which controls flicker and glare). This can be damaged by strong solvents.

Whether or not you feel the need to resort to specialist products to reduce dirt, grease or static around your monitor screen - it is a good idea to keep it clean. Screen resolution has been found to improve noticeably if a screen is cleaned after a few months or years of neglect.

A wise man advises: "If you are using anything even faintly liquid - turn your monitor off before cleaning it."



Open Channel

Machine code

I am very pleased with the series of articles provided by Alan Bridewell. These have been of great help to me in my endeavours to learn the art of machine coding. As I have no textbooks on the secrets of Qdos, these articles are very revealing. Hopefully there may be more to come.

Also, I would like to express my appreciation of the articles provided by Simon Goodwin, and the Quality of *QL World* as well.

**W H Starling
Plymouth
Devon**

Comment: We have been talking to Alan about doing some articles on QL machine code for complete beginners, as several people expressed a "start-up" gap. This is not a particularly easy venture, but we are working on it.

Balanced

I found the reviews of Minerva and Hermes by Ian Bruntlett very balanced. I agree with him about the Minerva manual, and we are re-writing it. Hopefully this version will be completed soon. Version 1.94 of Minerva will also be out shortly. (See *QL Scene*.)

A number of problems with old programs can be cured by turning off integer tokenisation. The difference in cost between MkI and MkII is primarily the cost of the extra I2C components and extra building costs. If the price was reduced, we would make less profit from MkII than we did from MkI - not very good business practice!

With Hermes, we now include all the MDRS

commands. The best features of the keyboard handling were not mentioned - keyboard with certain add-on keyboards is eliminated, but without affecting serial input. The manual refers to the correct LRESPR size for the toolkit. A few early Hermes were issued with the incorrect BOOT program - Ian must have got one of those.

On a different subject, Robert Taylor (Open Channel October 92) says that the PC versions of the Psion Four cost #495. He is talking about Psion *Exchange*, which allows task-swapping. His money would be far better spent on *PC Four*, which gives four stand-alone programs for #70 or so.

**Tony Firshman
TF Services
London**

Typical!

I just wanted to let you know that 14 months after you published the Desktop serial the program works. After more than a year I found a friend willing to check the listing for me, and he found that inside the GETSTRING procedure I had entered GUEST\$ instead of QUEST\$, which proves that when checking a typed-in listing you should not only use your eyes but also your brains. Beside this, he had found a bug in the 6th line of the TODOFILE procedure which should read: if ANSWER=1: ...

Anyway, I am now ready for another program by Mr Stevenson.

**Albert van Rheenen
Amsterdam
Netherlands**

Comment: about 90% of all listings problems are caused by mistypes.

Jargon

It's good to hear that *QL World* is alive and kicking. What I would like to see in the magazine are articles giving in-depth instructions as to how to get the most out of the software and toolkits available. In particular, I have in mind the utility software *Qram*

I have read and re-read the manual that comes with it, but I still can't understand what a hotkey file is, or what it does. Could I, for example, write a macro or Altkey to enable me to switch instantly from, say, *Quill*, to *Abacus*, without having to cycle round the whole lot of other programs I have loaded?

I don't consider myself stupid (I have even written a quite reasonable database program), but the manual of *Qram* is written in such a way that I feel it must be intended for some superior mortals with IQs in triple figures!

There are other features of the program which I don't understand, and this means I am only using a proportion of the facilities which it has to offer. This is only one program. There must be others which people have difficulty with, especially as QL owners are still in the position of upgrading to memory expansion and disk drives.

**Steven Hutton
Quanta Lancashire
Area Sub-group
Chorley
Lancs**

Comment: The ability to rub along with a computer manual has indeed convinced some members of the fraternity that they are superior mortals, but not very many, fortunately. Most people find most manuals very hard work and

thank the good gods if they can break the code. QL World always wants to hear from users who have a good grasp of any useful commercial or popular Public Domain QL program (including Qram - massive hint) and feel they can pass some of their experience on to other users.

Umlauts

In his review of Digital Precision's *Spellchecker* (September 1992), Bryan Davies wrote very little about the use of the Dictionary Utility (DU) to make a user dictionary. In fact, this is a very powerful part of the program.

About half of my wordprocessing is done in Dutch, and since receiving *Spellchecker* I have been able to write a Dutch dictionary of over 23,000 words. The DU program is easy to use, and, even with a dictionary this large, adding 200 - 300 words to the dictionary is completed well within half an hour on a Gold Card QL. Bearing in mind that 23,000 words have to be sorted before the dictionary itself can be built, I find this an impressive speed.

There are, however, two shortcomings of user dictionaries which are especially disadvantageous when attempting to write a foreign language dictionary. Firstly, the maximum word length in a user dictionary (unlike the supplied dictionaries) is 17 letters. Like German, Dutch has a liking for long compound words. However, I find that this is only a serious problem in very academic texts, and I understand from Freddy Vachha that DP may

eventually address this shortcoming.

Secondly, DUU ignores letters with an accent or an umlaut. Fortunately, Dutch has relatively few words with accents and umlauts, and I can get away with adding these to the dictionary as two words. (For instance, "client becomes "cli" and "nt"). Nevertheless, this shortcoming would be more serious if you attempted to write a French, German or Scandinavian dictionary.

Are there any other readers with experience of writing large user dictionaries?

Geoff Wicks
Amsterdam
Netherlands

Comment: We have heard from one German user who is unhappy with the way that his Perfection responds to umlauted words when spellchecking. Hopefully DP will issue some help to users running into problems with accented words. Umlauts and accents give QL World gyp when file-converting to the office Macs, as well, so QL users are not alone.

Unwired?

I recently purchased a second hand (but unused) Sinclair TTL monochrome monitor model S-12MM. I am having difficulty is working out the connections from the 8-pin plug at the computer end to the D-type connector.

Over the years there have been similar queries in *QL World* so I went through my backnumbers to see if they would shed any light. I found the items I was looking for, but have had no success in getting the monitor to work. The screen lights up when I switch on, so I assume it is the connections to the D plug.

Six pins only of the plug are used, as follows:

1 red, 2 black, 6 white, 7 blue, 8 orange and 9 yellow. The red wire on pin 1 is less than one inch long and doesn't seem to do anything at all.

Does anyone have the appropriate experience of this particular monitor, or know of anyone who could advise me of the correct wiring to get it

working?

Although I use my QL almost exclusively for wordprocessing, I find *QL World* a most informative magazine.

C F Marsh
Cheltenham

Comment: I have made some enquiries, but my enquirees don't know their way around that monitor. We need information from someone who does - please contact Open Channel if you can help.

Paging QL

You ask for comments on the new page style. Ragged right - yes, indeed. Every time. Not just because it eliminates the redundant spaces that fret Martin Harris so, but also because in a multi-column page they are not so much eliminated as added to the "official" column separations.

Nose down? On the principle that one follows one's nose, are not most of your headlines nose up? One function of a headline is to separate succeeding from preceding text. This has not happened between pages 32/33.

In Geoff Wicks' article on page 35, he refers to the practice of viewing an inversion of a page to judge its effect. When I was an apprentice in the shipyard, drawing plans on transparent paper or cloth, I was taught that a powerful method for error detection was to turn one's work over and view it from the reverse side. Or even take the thing up to the photographic department and get them to run it through their machine the wrong way up.

P H Tanner
Glasgow

I'd like to congratulate you on the new look of the magazine. The layout is certainly more readable, and there seem to be far fewer typographical errors. Perhaps the Grauniad should take note?

Alex Munden
London

Open channel is where you have the opportunity to voice your opinions in Sinclair QL World. Whether you want to ask for help with a technical problem, provide somebody with an answer, or just sound off about something which bothers you, write to: Open Channel, The Blue Barn, Tew Lane, Wotton, Woodstock, OX7 1HA.

While I have no strong views about ragged-right, I definitely do not like the so-called nose-down headlines. I found it irritating, having to turn the mag sideways.

I would also like to take this opportunity to comment on the magazine wrapper. There is an exhortation to the Post Office that, if undelivered, it should be returned to the sender. The sender's address does not however appear! Apart from the addressee, the only visible address is that of Dilwyn Jones!

Stuart Johnson
Peterhead
Aberdeenshire

Editor's comment: I personally prefer justified text, but find the ragged columns, (apart from very

narrow ones) easy to read and spacious, and the vertical headlines useful for flipping through the pages. The number of typos has dropped because I now have a stranglehold on the text at all stages, and more opportunity to check it. Foul-ups are now less numerous, but more spectacular. Putting our address on the wrappers would mean the cost of another labelling process. The Post Office (unlike the US Mail) should check inside returned post for the sender's address, but they usually adopt the simpler measure of sending it to Dilwyn.

Reversing a layout to see if it is working is a sound procedure, but the thought of turning one loose in a shipyard drawing office is a little disturbing!

Editor's notebook

Stop press: Freddie Vachha of Digital Precision contacted me to say that C-Libraries, samples and toolkit support will now all be bundled with CPORT (see page 18) and not only with C68.

Do you go to All Formats Fairs or Quanta meetings? If not, why not? Answers on a postcard, please ... Reader John Reid requests a regular "meetings diary" in QL Scene. We'll tell'em as often as we find'em! A bit more notice from Quanta would assist in some cases.

November's QL World followed the pattern of the October QL World and came out at the end of the month, for the same reason. I am hazarding a guess that December's one will follow the same pattern, so I have two things to say about that:

The first one is: the readers tell me that being late makes the advertisers unhappy, but the advertisers tell me that it only makes them unhappy if it makes the readers unhappy. What everyone wants is to know when to look forward to their QL World. That's what we are working on. Don't worry about your Readers' Survey - we're allowing extra time for it.

The second one is: Happy Christmas! And a merry New Year too.

**The first QL, the angels did say
Was the start of a habit
That's stuck till this day!**

QPACER

**Qpac 2 user
Henry
Orlowski
investigates
a new way
of setting up
boot files.**

INFORMATION
Program: QPacer
boot-writing utility
for Qpac 2. Needs:
SuperToolkit 2,
memory expansion,
3.5 in disk drive.
Price: Shareware
Publisher: Quanta
Library

Qpac by Jeremy Davis is for people who either have Qpac 2, or already have Qram and are considering an upgrade. These programs provide a system control environment to enable you to manipulate your files and programs in a relatively straightforward manner, and in so doing provide a multitasking structure. If that sounds good to you, get Qpac 2 and then consider Qpacer. If you've already got Qpac 2

to configuring the program to your own personal requirements. If you cannot write an effective boot for yourself then without doubt you are not getting the full value out of the program.

Manual

Fortunately many Qpac 2 users realise this, and try to improve their knowledge and skills. Unfortunately, not all succeed. The manual is difficult for many people to follow and grasp. The availability of Qpacer could be a timely introduction.

Qpac 2 needs QL users who know what to do with a QL. Not all QL users have the knowledge to get stuck in to such an extent, and are therefore disappointed by Qpac 2. This is a shame, for it is well known to be an excellent piece of software. Qpacer aims to provide that missing link.

Writing the boot for your Qpac 2 is critical and difficult

How should they be presented?

In what form are the applications to be held?

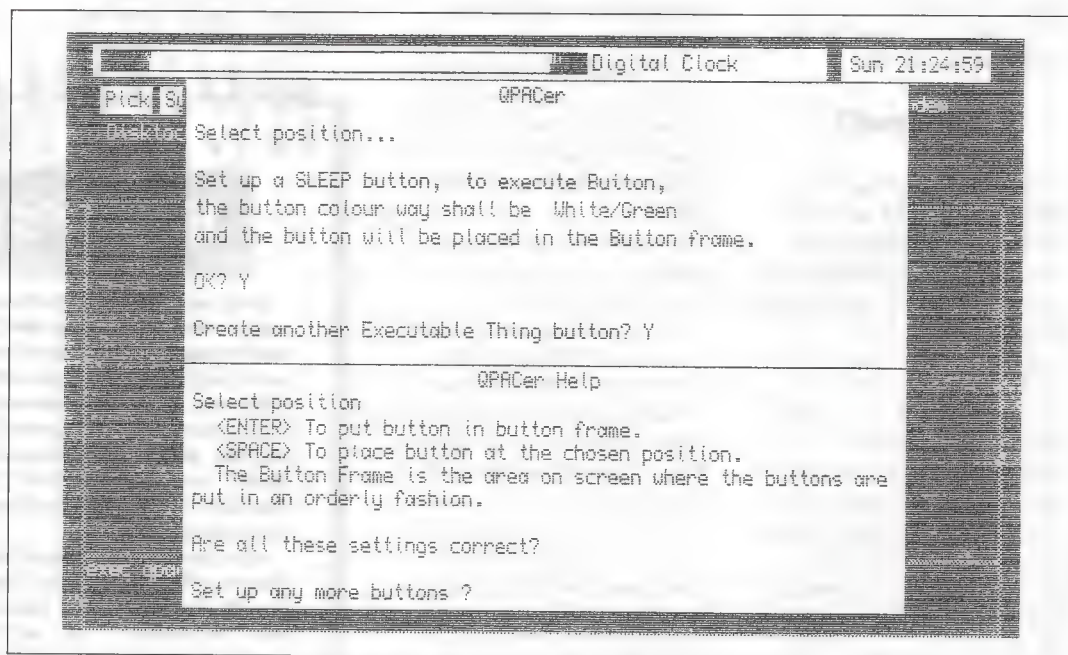
How are they to be called?

Answers

Once you have answered these questions you are on the road to getting the customised user interface that your boot provides for you. Qpacer is designed to get you there with greater ease.

It's quite simple to create a working copy of Qpacer. You either use the copying program provided, or copy files via Superbasic, ensuring that you have the relevant Qpac 2 files themselves (ptr_gen, wman, hot_ext, Qpac 2), to create a joint Qpac 2/Qpacer disk.

Then run the program and answer the questions it asks. Program operation and use is very straightforward. As well as asking you questions it constantly suggests answers. The answer available for



you may already know all about 'buttons', 'hotkeys', 'executable things' and other such goodies; or you may have read about them in the manual, but not really know what they are or what you're supposed to do with them. Whether you feel confident with these things or not, you may well have difficulty writing a Qpac 2 boot file. Boots are very important in Qpac 2, because they are the means

for the same reasons. Whereas Qram provided something of a fixed front-end for system manipulation, Qpac 2 needs much more of a personal feel about it. It's the boot that gives this critical personalised 'feel'. It's difficult to write the boot, because you have to consider a number of important factors:

Which items should be included in your set-up?

selection is highlighted for you to accept, or you can move or toggle between the options, leaving very little for you to type in yourself. The structure of operations is precise and methodical, as it needs to be for something as relatively complex as a Qpac 2 boot. There is constant context-sensitive help on screen throughout, with many succinct explanations of the various Qpac 2 options.

Confirmation

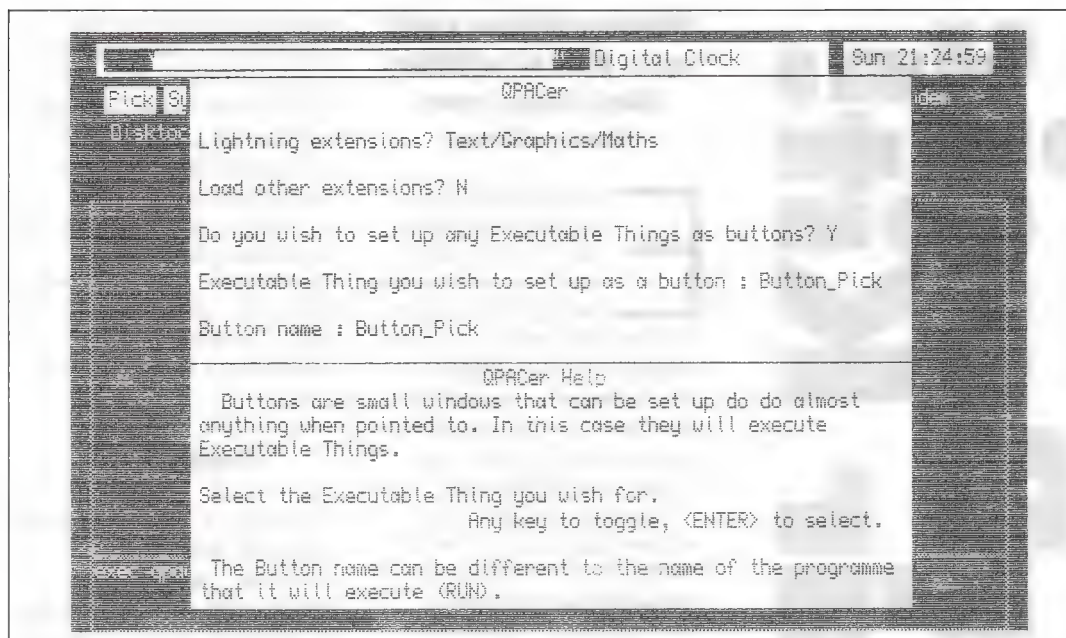
The program moves from System extensions, to Hotkeys, to Buttons, to the Final Outcome of your boot. It's also error-trapped, not allowing you to overwrite boot file names, for example, unless you confirm that you want to. Finally, before it writes a section into your boot it lists the selections you have made and asks you to confirm your acceptance. This is extremely useful as it's very difficult to remember exactly the string of choices you've just made.

After you've settled on a name for your boot, the first thing you are asked is whether you wish to load any extensions, such as Lightning, Turbo Toolkit, or Qload speed loader. Select the item you want. In the case of Lightning you choose whether you want Text only, Graphics only, Text and Graphics, or Text, Graphics and Maths.

Throughout each input section the program asks you whether you wish to have any more selections included, before it moves to the next option, Buttons, small windows which are visible for easy selection of specific operations. Buttons can be one of three types: Sleep, Wake and Exec. The differences are quite subtle and relate to the way in which the applications are available and how many concurrent copies can be in memory. If you don't know the difference, consult the Qpac 2 manual, or as an alternative consult the Qpacer manual (available as a .doc file on the master disk), which gives a compact explanation.

Suggestions

Of course, you have to type in the name that is to appear on the button. Qpacer won't suggest this for you, but suggestions for the remaining options are mostly available on screen for selection. The first of these is the colourway, a choice of four different main colours and outlines. Different groups of applications can be given different colourways, for instance, white and green for



pop-up utilities, and black and red for wordprocessor packages. You then have to select where on the screen the button is to appear, either by moving a flashing block or just pressing Enter to put it into the Button_frame, which automatically positions the button for you in the next available space. The Button_frame is the better option; otherwise you can easily get lost. The program does not remind you of previous positions and you end up with hidden or partly-hidden buttons which look very inelegant.

Create as many buttons as you like before moving on to Hotkeys, combinational keypresses similar to the 'altkeys' of TK2, designed to implement a specific action. Select the type of Command hotkey or the type of Resident hotkey that you want. Again, the manual explains the differences. Confirm whether you want to 'call' it once only or not, and give it the name of the file it relates to.

Things

Then set up your 'Thing' hotkeys to include such Things as the 'files' menu, the 'button_pick' thing or the 'button_sleep' thing. The program highlights all the Qpac 2 Executable things to make it as easy as possible for you.

Next, decide whether you want any Pick or Wake hotkeys, and give them a 'job

name'.

Finally, you can choose to set up your hotkey as a Button. If you do this, you can forget the combinational keypress, just point to it and the system will do the rest for you. You will be asked for the colourway, the name and the position as before.

Once you have finished entering all your hotkeys you are asked what outcome you want once your boot has been run, in other words where you want to start your day. This could be at SuperBasic, or you could load a program of any type, or execute a Hotkey, or a Thing, or indeed turn SuperBasic into a Button. If you select SuperBasic, you can at the same time choose to set the time and date if you want, especially if you do not have a real-time battery-backed clock.

Remember, however, that at this stage you can just keep going, setting up as many Hotkeys and loading as many files as you want before ending up with SuperBasic on top of the pile.

Another?

Once you have finished your boot set-up, you are then asked if you want to 'create another boot?' and the whole process can start over again.

The program can be configured for a variety of default options. These include the Device, the Directory, the User Level, the Button Colourway, the Year, the

Month, Start line number and Increment.

The User Level can be Novice or Competent. The Competent Level gives more options, for example, on the device and directory names, the colourways, Files menu configuration, Command string and Control Code options.

Defaults can also usefully be set at 'exec Qpacer' time by adding a simple command line to the instruction, saving a great deal of time and allowing you to run different versions of set-up as required.

Qpacer is indeed a useful program. It doesn't do everything, but what it does it does very well and with maximum simplicity. However you may still have to doctor your boot before you run it to make it do exactly what you want it to do. It doesn't give you the option of reducing SuperBasic's windows, for instance, although this seems a relatively easy item to incorporate compared to some others. It doesn't 'remember' where you put your buttons unless you put them in the Button_Frame, so you might have to twiddle around independently of Qpacer to get your positions just right. It would also have been nice to have an option of seeing what you've got so far, as you go along.

But you can't have everything, and all in all I have seen less useful programs sold for good money while this one is virtually free.

Perfection

Shirley Butler calls up a trio of utilities that keep your Basic clean.

Every Superbasic programmer should have some tools in their diskbox. I'm not talking about the old trace utilities, where you run a program with the executing line displayed somewhere on the screen. Trace programs can be handy but they are limited in what they do.

Take this situation: You wrote a large program a few months ago and tested it as thoroughly as you could. Now, suddenly, you find a bug in it, or you want to do some modifications or updates to it.

It's ten to one that you can't remember every variable and string array you've used. You've probably forgotten the procedure-names that you have defined, and what kind of returns the functions make. What do you do? Start looking through the program and unravel it?

A program guide

No. You load in *Xref* and let it do the job for you.

Xref will give you a breakdown of the program, and show you the variable types as they are defined, dimensioned, assigned and used within the program.

It also gives a list of all the procedures and functions you've used, and how many times they have been called.

The above information is a good place to start when you are trying reacquaint yourself with your Superbasic program, especially if it is a large one with many variables and procedures.

The information that *Xref* can supply does not end there. Your attention will be drawn to potential problem areas. Procedures and variables that have been defined, but not used, are displayed. This helps you to 'cut the slack' in your programs.

Redundant parts of programs, and stray arrays, variables and procedures can be left lying around,

especially when you are updating and modifying the program. Practically every program that I have written has been put through *Xref* and *Better Basic*, and has benefitted from it.

Xref is an analyser utility for Superbasic programs, whether they run under the interpreter, compiled with Turbo, interpreted with Qliberator or translated with Cport.

Using Xref

Is it easy to use? The simple answer is yes.

You can select *Xref* to output to the screen or printer. If you select screen, then all output goes to screen. Should you select printer then you have three options, serial port 1 or 2, or parallel.

If you select parallel, you can type in a device name like par or par_12k, or you can spool the output to a file by typing a filename something like flp1_prog_report. This will allow you to examine the *Xref*

analysis at a later date with an editor, or a program such as *Perfection*, and put it to printer if you wish.

You can also select *Xref* to produce either a full or exceptions report. The

exceptions report causes a full analysis on the program, but will only produce a final report on potential problem areas. The full report will allow you to select the procedure analysis option. This will trace all procedure and function calls and establish program hierarchy, creating a more detailed report.

During analysis, *Xref* will display the current line being scanned. If it finds error lines containing the Superbasic keyword "MISTake" then the buzzer will sound, and the line number will be displayed.

If you are a Superbasic programmer, serious or otherwise, it is well worth having *Xref* lurking in your

disk box.

Xref will analyse your program. But what happens if you need information about where to look to polish your Superbasic code, to make it faster and/or more efficient?

Monitoring

The answer to that could well lie in *Superbasic Monitor*. This is a utility that sits in the background and follows your Superbasic program as it runs, sampling the time spent in different parts of it. Frequency of sampling, which is the number of times that the monitor takes a snapshot of the program, is selectable by the user.

Superbasic Monitor will then turn the sampling into percentages of how long is spent in each part of the program. The parts of the program with the highest percentages will be the ones likely to need pruning or tidying.

In the report produced by the monitor, the line number range used for sampling is

But what happens if you need information about where to look to polish your Superbasic code, to make it faster and/or more efficient?

displayed in two columns: the first "from", the second "to".

Percentages for that range are also displayed in two columns.

The first column of the percentages is based on the

net frequency, excluding the count for the idle time in the Basic program.

The second column of the percentages is based on the total frequency, including idle time.

Superbasic programs that require the user to "press a key to continue" can spend quite a lot of time doing nothing, hanging around inside a few lines of code, waiting for a keypress. If it is possible to omit these lines or encapsulate them in a REMark, a more meaningful report can be obtained for the overall program. This is the name of the monitor's game.

There is one problem that could arise. The RESPR

INFORMATION
Programs: *Xref*
Superbasic
analyser
£29.95)
Superbasic
Monitor
(£24.95) *Better*
Basic (£24.95).
Supplier:
Digital
Precision, 222
The Avenue,
London E4 9SE.
Tel. 081 527
5493.

command will not operate if there is another task running apart from Superbasic. If your program uses RESPR, it is best to use it before running the monitor. The Turbo Toolkit's ALLOCATION command can be used in place of RESPR, and will work if another task is executing at the time.

Using Monitor

Superbasic Monitor is simple to operate and there are no extensions to load. Simply load in your Superbasic program, issue the direct command EXEC flp1_sbmon, answer a few simple prompts and then run the Superbasic program that is laying in memory. The monitoring of your program is automatic, and you can look at the report produced at anytime with a simple keypress.

If speed and efficiency is on the list of your programming priorities, then you should find this program is a good one to have around.

One programming utility that I use a great deal is *Better Basic*. To be honest it is something that I would not want to be without, and it is my particular favourite (and most used) of this trilogy of utilities (*Utility? Trility? Ed.*)

One of the advantages of Better Basic is that you can neaten your Superbasic source code as you wish, with a minimum of fuss or work. Reformatting your program in the form of indenting and renumbering, and automatic linelength - which can allow the turning of multi-statement lines into single statement lines. What can this achieve?

For a start it can make your programs easier to read. It can also make them easier to modify and maintain, and a good structure is what is needed to do that. If that is what you require, then Better Basic is worth it on that count alone.

END IFs, END FORs and END DEFines are added if the programmer has left them out. Noise words such as THEN and LET are stripped out. They are not needed in Superbasic, as they are just overhangs from other Basics,

included by Sinclair to make programmers (of other computers) new to the QL feel more at ease.

BB comments

You can select to allow Better Basic to insert comment lines. This will apply to BB's addition of missing END IFs and so forth to the Superbasic source.

BB's insertion of error messages cannot be suppressed and will be included regardless of your selection to include comment lines. The error messages are various, clear and to the point in order to draw your attention to them.

An error and warning count is displayed on the screen and updated as warnings and errors are found, so you will know how many of each there are to look for and correct after Better Basic has done its stuff.

END IFs and such can be selected for reformatting in a couple of different ways. As an example, the following Superbasic program is not very neat or interesting to look at, and it could be easier to read as far as the IF and END FOR statements are concerned. It can be reformatted and indented into something much better.

```
10 Do_Something
20 DEFine PROCedure
Do_Something
40 FOR x=1 TO 10:IF
x=5:PRINT x:END IF :END
FOR x
80 END DEFine
Do_Something
```

Better Basic reformat 1 (below). This shows one kind of indenting which certainly is easier to read than the program above.

```
100 Do_Something
110 DEFine PROCedure
Do_Something
120 FOR x=1 TO 10
130 IF x=5
140 PRINT x
150 END IF
160 END FOR x
170 END DEFine
Do_Something
```

Better Basic reformat 2. I prefer this. The indenting is

only subtly different from reformat 1. It does seem more logical, though, in that the END terminators for the corresponding FOR and IF statements are in the same respective columns. To me, this makes the program stand out clearly. It also makes it easier to read, understand and update.

```
100 Do_Something
110 DEFine PROCedure
Do_Something
120 FOR x=1 TO 10
130 IF x=5
140 PRINT x
150 END IF
160 END FOR x
170 END DEFine
Do_Something
```

During the process that Better Basic carried out, some helpful messages appeared on the screen suggesting the use of NEXT and EXIT in the FOR and END FOR loop - which could actually be used to cut down on processor time.

98 warnings!

I have one particular program that has been an ongoing concern for the last few years. During that time it has been re-written, updated, modified and reformatted by hand.

I decided it might be interesting to put it through Better Basic and reformat it properly. It is a program that I thought was bug-free. Better Basic reported 98 valid warnings and one error in the program, which goes to show how wrong one can be.

Better Basic corrected the parts of the program that caused the warnings and called my attention to the error with a polite message.

I frequently use *Turbo* and *Cport*, both also published by Digital Precision, and I find that Better Basic is a good accompaniment to them. Cport especially is less tolerant than Turbo and requires a Superbasic source to be correct or more correct than Turbo does. There are some things Turbo will allow that Cport will turn its nose up at.

Cport will report an error where Turbo will do an auto

correction where it can - Better Basic can help greatly to minimise this error flagging.

Don't go away with the thought that Better Basic is only for people who "can't program". It is a boon as a cleanup/tidyup utility for any Superbasic programmer

Chas Dillon

The above three programs are all well laid out, written in Superbasic and compiled with Turbo - by Chas Dillon, who is an excellent programmer. All options are handled with simple prompts and explained in their respective manuals.

The documentation that accompanies each of the programs, in this trilogy, is well laid out and informative.

**The manual for
any program is
worth reading
fully in order to
get the best out
of it, and it is so
in this case.**

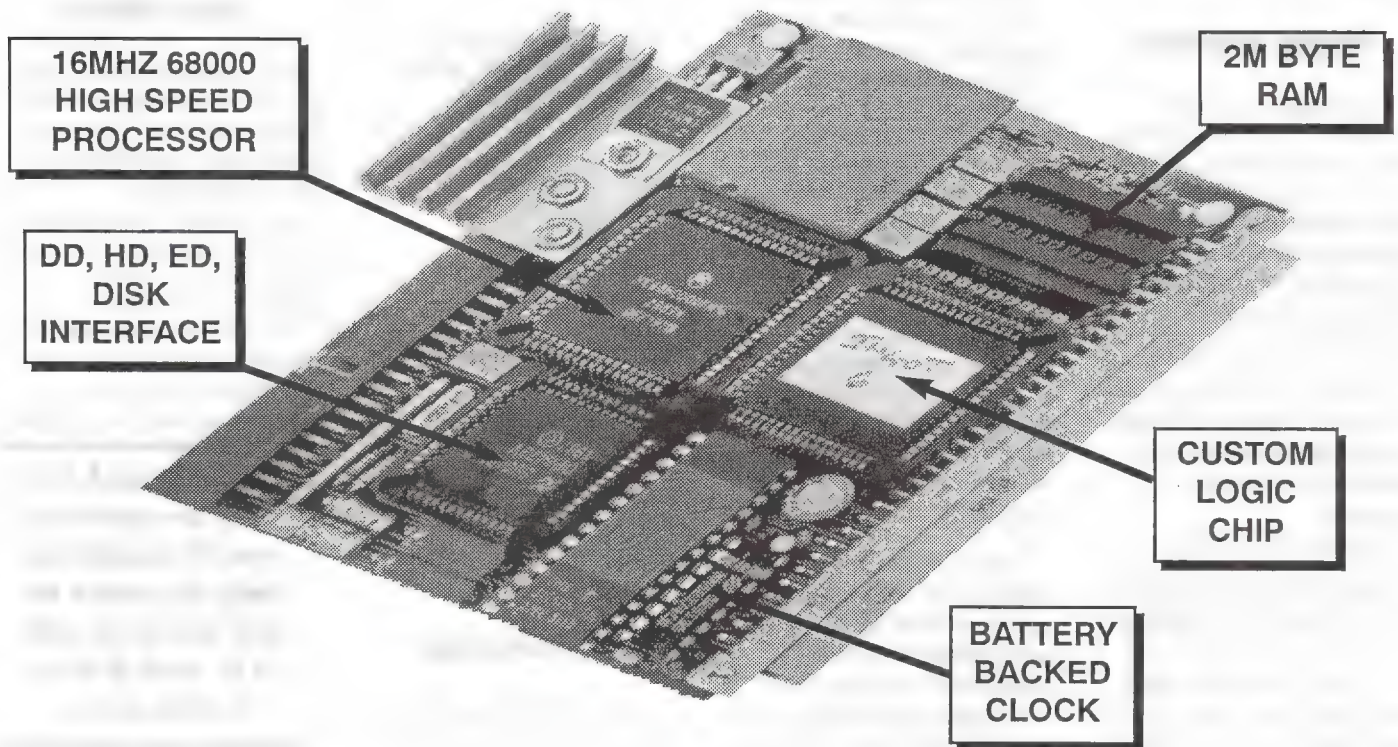
The manual for any program is worth reading fully in order to get the best out of it, and it is so in this case. I have quite a collection of software from Digital Precision, and I have never had any problem in understanding any of their program documentation - and I regard that as a plus.

I use BB whenever I program in Superbasic. The correcting, reformatting, renumbering, checking structures and warning/error flagging makes it an ideal utility for the Superbasic programmer to have. It leaves me wondering "How did I manage without it?"

I think of *Xref*, *Superbasic Monitor* and *Better Basic* as my Superbasic repair outfit.

**SuperBasic
Repair Kit!**

MIRACLE



QL GOLD CARD

£225 inc. (£200 export)

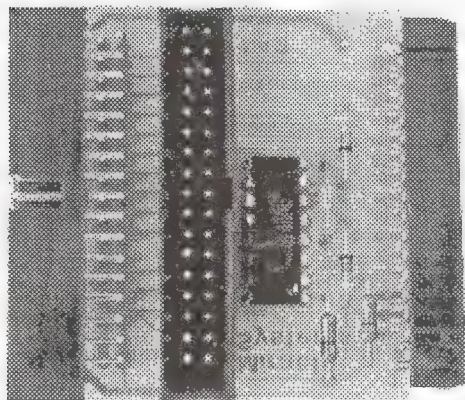
This is the expansion that has been revolutionising the QL. It is very easy to fit - it simply plugs into the expansion port at the left hand of the QL - and once fitted it will instantly increase the execution speed of the QL by about 4 times due to the presence of a 16MHz 68000 on board. There is 2M of fast 16 bit RAM of which QDOS sees a contiguous 1920K. The remainder is used for shadowing the QL's ROM and display memory and for the GOLD CARD's own code.

There is a disk interface which can access 3 mechanisms (4 with the DISK ADAPTER) of 3 different densities, DD (double density, 720K), HD (high density, 1.44M) and ED (extra high density, 3.2M) in any mix. The disk interface connector is the same type that was fitted to the TRUMP CARD so most QL compatible disk drives can be used. Please note that DD drives still give a capacity of 720K per diskette. Our DUAL ED DISK DRIVE allows the GOLD CARD to access DD, HD and ED diskettes.

Another feature is the battery backed clock. When the QL is switched on the contents of the clock are copied into the QL's clock so that the time and date are correct. The firmware in the ROM gives the GOLD CARD all the functionality of the TRUMP CARD like TOOLKIT II and there is a sub-directory system for floppy and RAM disks.

Physically the GOLD CARD is about half the size of the TRUMP CARD and so fits almost all within the QL. Its current consumption is well under the allowable maximum so no special power supply is required. The GOLD CARD comes with a 14 day money back guarantee and a 2 year warranty.

SYSTEMS



DISK ADAPTER

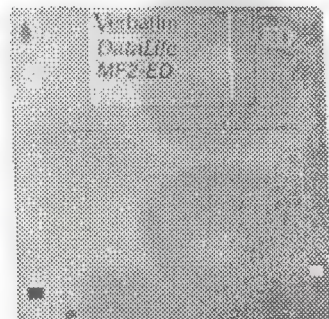
£15 inc. (£15 export)

Plugs into TRUMP CARD or GOLD CARD to allow access to 2 dual disk drives (i.e. 4 mechanisms) as FLP1_, FLP2_, FLP3_, FLP4_.

BOX OF 10 ED DISKETTES

£30 inc. (£30 export)

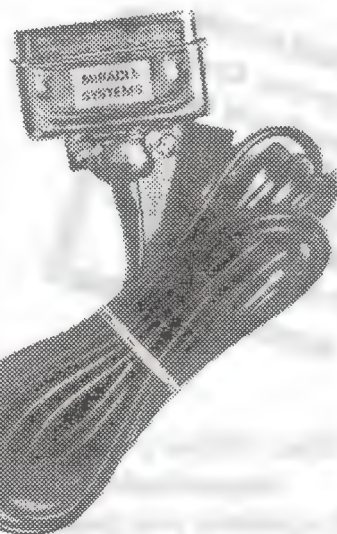
Ten 3.5" ED diskettes. Gives capacity of 3.2 MBytes when used with GOLD CARD and DUAL ED DISK DRIVE.



QL CENTRONICS

£25 inc. (£25 export)

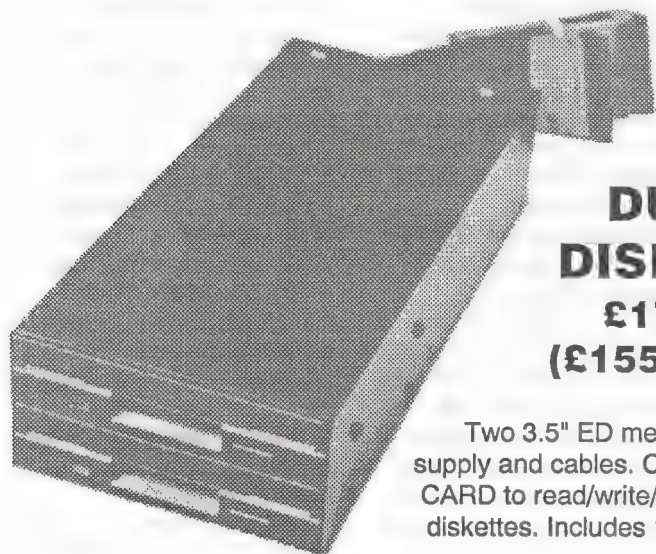
Connects SER1/SER2 to Centronics parallel printer. 3 metre cable included. (D9 serial plug option for German and American QL available.)



DUAL ED DISK DRIVE

£175 inc. (£155 export)

Two 3.5" ED mechanisms, power supply and cables. Connects with GOLD CARD to read/write/format DD/HD/ED diskettes. Includes 10 ED diskettes.



Tel: (0904) 423986

To place an order by phone please have your credit card ready. For overseas customers we charge the prices shown in brackets.

To order by post, please fill in the form opposite or write to us quoting your credit card number and expiry date, or enclosing a cheque payable to MIRACLE SYSTEMS Ltd.

To: MIRACLE SYSTEMS Ltd, 25 Broughton Way, Osbaldwick, York, YO1 3BG, U.K. Tel: (0904) 423986

Please send me _____

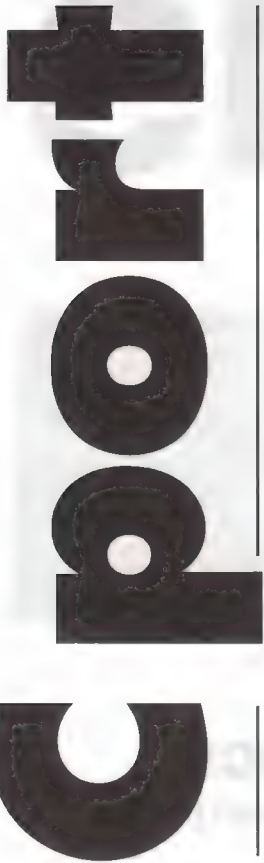
I enclose a cheque to the value of: £ _____

Or debit credit card

Expires /

Name _____ Signature _____

Address _____



INFORMATION
Program: Cport
SuperBasic to C
translator
Supplier:
Digital Precision
Price: £89.95

Chas Dillon's Cport
SuperBasic
translator has been
re-issued by Digital
Precision. Shirley
Butler is a fan.

The programming language C is a powerful one, allowing many of the advantages of machine code at a higher-level programming.

Normally you would write a C program using a text editor. To run such a program it must first be compiled into an executable machine code program with a C compiler.

One of the brownie points

that C achieves is that it is a portable language and can be transferred to compile on another computer. The C source may need to be modified to some degree to enable compilation once transferred.

A lot of people would like to take advantage of such a language, but don't because they cannot be bothered to learn it, or just cannot get to grips with it.

Basic to C

Consider, then, a program that would take in a Superbasic program and churn it out as a C source? Superbasic is similar to C in that they are both structured languages, and so should lend itself well to such a translation. If such a program existed, would you use it?

Such a program does exist. It is called *Cport*, and is available from Digital Precision, written in SuperBasic by Chas Dillon, and compiled with *Turbo*.

It's pretty easy to use and it does the job well, and, considering the amount of work it does, it is fast. It requires a QL expansion of at least 256K to run and can be used by someone who has no knowledge of C at all. Having said that, however, in my opinion some knowledge of the language is desirable - but then again, knowledge can be picked up along the way.

Cport has an identical menu form to the Turbo Compiler, so Turbo users should feel at home with it straight away. To the uninitiated it is an easy menu system to get used to.

Cport can be used in an educational way to learn about C. For instance, type in some Superbasic, Cport it, and study (and tinker with) the resulting C source - much can be learned from studying programs to observe how they work.

To study the C source that Cport produces, you can use one of a bevy of editors available for the QL. These include *Editor*, *Perfection*, *Spy* and *QED*. Personally I use *Perfection* or *QED*. Word processor programs such as *Quill* are not really suitable for this purpose.

Translating

Cport is a translator, which is completely different from a compiler or an assembler. A translator is an extremely intricate program to write, especially if it is to translate the whole syntax of one computer language to another. Cport is a huge project in programming terms and is, I believe, the end product of four years' work.

Cport is not a magic wand to make incorrectly written programs, that won't work in SuperBasic, into programs that will work once translated into C. Your SuperBasic source must run correctly under the interpreter in the first place.

Cport has a number of options available to the user for handling the translate process. There is not room to explain them here, but the manual that is distributed with Cport gives a clear description of what each option does. I have read the manual, and there does not appear to be anything that cannot be understood with attentive reading.

To turn Cport-generated C source into executable machine code programs you will need a C compiler, CfiX and Cport support libraries.

C68 is the recommended C compiler to compile the C source generated by Cport. If you believe that obtaining a good comprehensive compiler is going to be an expensive job then I am pleased to disillusion you.

C68 is a Public Domain program, produced by David Walker and distributed on three disks for well under a fiver. The strength of any C compiler is in its set of libraries. C68 has an extremely good set. I use C68 myself; it is the only C compiler I have for the QL and I'm so pleased with it that I do not feel the need to search around for another one.

Although it is recommended for use with Cport, it is an excellent ansi C compiler in its own right for raw C programming and represents very good value for money, especially as QED, a fully functional Ascii text editor, is also included - a complete C programming environment in one package. C68 can be bought with Cport from Digital Precision or from a QL Public Domain outlet.

Library calls

When Cport translates, it turns some of the Superbasic keywords into library calls. To handle these calls, LibCport, an extensive set of special libraries (also written by David Walker) is available separately from various Public Domain sources. You could of course write your own libraries, but to do that you would need a vast knowledge of C and a great amount of time. The libraries on the LibCport disk were written in C and compiled with C68.

Cport inserts intelligent and helpful comment lines into the generated C source. Some of them refer to errors and redundant DIM statements, and some are warnings concerning variables that have not been declared globally or locally. So it is worthwhile looking through the C source to see if you have any warnings that need to be dealt with. Warnings concerning LOCAL/GLOBAL variables can be dealt with in two ways: (1) by editing the C source (2) by editing the Superbasic source and then Cporting again.

During the time that Cport processes a SuperBasic source it produces two files to be used for compilation.

File one has `_c` appended to it. This is the C source and contains the Cport translation of the Superbasic program.

File two has `_h` appended to it. This is a header file and contains prototypes of user defined procedures and functions used in the C source. Also in the header file is a table of all procedures and functions that are resident in the QL, or have been loaded as SuperBasic extensions - *Toolkit II*, *Turbo Toolkit*, etc. At the end of the header file is usually a list of `#INCLUDE` statements. These have been made redundant by the creation of the LibCport support libraries and can be removed or disabled.

Errors

I pull no punches here. There are some errors that Cport makes in the translation process. It is inevitable that this happens; no translation can be absolutely pure. The errors are

no problem if you have a knowledge of C or have some guide to how the errors can be put right. On the LibCport disk there is documentation that will help you.

The errors are not serious and are easily corrected if you know what you are looking for, but some errors are subtle syntax mistakes and could be prevented in the Basic source before Cporting. These errors, although a very small percentage, will however prevent the C source from compiling.

Repairing the errors that Cport makes is more time-consuming than difficult. Bundled with Cport is a program called CfiX, which will do the work for you. It will examine the C source produced by Cport and make the changes and additions to the source to allow it to compile.

The job of finding the errors and repairing them in a long program could take anything from fifteen minutes upwards. CfiX is easy to use, speedy, and can do it in a fraction of the time that it would take a human to do it manually. 700 lines of C would take CfiX approximately 1.75 minutes to correct, depending on the amount of work it has to do.

Cport can produce both Lattice and Ansi C source. As C68 is strictly ansi and the recommended compiler to be used, it is suggested that you set Cport to translate to Ansi C.

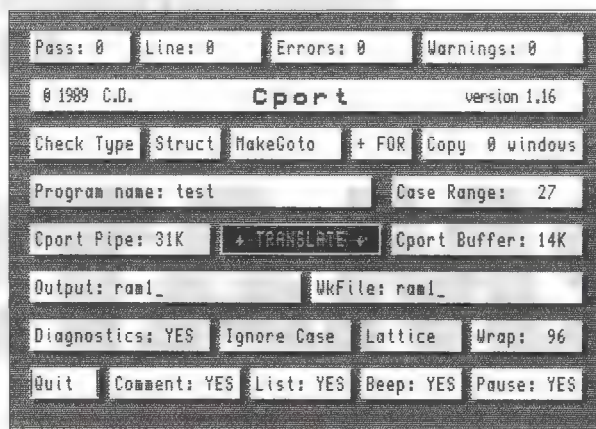
There are things required by the programmer when producing a Superbasic program for translation by Cport. Dimension all string arrays. Declare numeric variables with the Turbo Toolkit IMPLICIT directive. Declare LOCAL variables properly, don't just invent variables in the middle of programs willy nilly - Turbo will tolerate it but Cport won't.

It is reckoned that a program that compiles cleanly with Turbo has more chance of being cleanly translated with Cport. To a point this is true but, and there is always a but, in certain circumstances Turbo will make corrections where it can, to produce a clean compilation of the Superbasic program. Cport may not be able to make the same corrections.

For instance, during

translation, Cport will alert you to an undimensioned string array, but do nothing about it.

compilable. Since I have used Cport I can honestly say that my knowledge of C has



Turbo would dimension that array to a default value of 100. Turbo can probably cope with such a situation as it has its own code generator to produce the final executable machine code. Cport does not have this facility and has to rely on a separate compiler to produce the final code.

Any undeclared numeric variables in the SuperBasic source will be translated by Cport as floating point, which are inefficient and memory hungry in C. It is best to use integers where ever possible, and this is easy in SuperBasic with the Turbo Toolkit IMPLICIT% command - Cport will translate such declared variables as short integers.

Cport itself is a marvellous program, despite the fact that some errors do crop up during the translation process. Do not allow the errors to detract from Cport itself. Having used the program to translate SuperBasic into C, using CfiX to auto correct the C source, and then successfully compiling to a fully running executable program with C68, I can say that the process has been with the minimum of fuss. In fact I have written several programs in Superbasic which have been Cported, CfiX'd and compiled without any manual correction on my part whatsoever - *Towers Of Hanoi* on the LibCport disk is one such program.

Personally I like Cport, I use it as much as I use Turbo - and that is a great deal. The errors that Cport generates are no problem, with CfiX available. In general the syntax of the C source is correct and

increased a great deal.

I haven't just been issued with a copy of Cport, played around with it a bit and then put my thoughts down onto a word processor. I bought a copy at the January 1992 computer show at the Horticultural Halls in London. Since then I have made constant use of it. My experience of it leaves me with the conclusion that trivial to serious Superbasic programs can be Cported and compiled. I have even had such programs running on a Public Domain QL emulator on the Commodore Amiga.

In my opinion Cport is a viable proposition as a C programming tool. It is possible to write routines in SuperBasic, test them, Cport and CfiX them, and then lace them into an independent C source.

Cport will live in harmony with the Trump Card, Gold Card and Atari computers with a QL emulator. I have witnessed it running in all three situations with no problems.

Since Cport was first released by Digital Precision, a couple of independent programmers have worked extremely hard. One to create and develop the necessary Cport support libraries. The other to create CfiX to automate the correction of the C source, with as little manual intervention as possible on the part of the user.

You may not know C at the moment but you could well find that buying Cport will set you on the road to learning it.

There are some additional programs that you will probably find useful to use with

Superbasic programs before Cporting them. *Better Basic*, *Xref* and *SuperBasic Monitor* are three of them, and are available from Digital Precision.

I previously mentioned some extra programs that would be required to make use of the Cport generated C source code.

C68 Compiler on three disks... Runtime disk - the C68 compilation system. Runtime documentation - C68 documentation. Runtime utilities - various utilities plus their documentation. C68 was written in C and actually compiled itself. The source code for the C68 compiler is also available on a series of disks. C68 is a huge project in terms of programming and is subject to periodical updating and modification.

LibCport support libraries... These libraries were written in C and developed by David Walker and compiled with C68. There is no guarantee that they will be compatible with any other C compiler other than C68. CfiX is also shipped on the LibCport distribution disk.

CfiX auto corrector... CfiX is an auto-corrector / converter program for use on Cport generated C source. It is now included in the Cport package and with the LibCport support libraries. It is also available separately if wished. It is only of use in conjunction with Cport and C68.

Both CfiX and LibCport are subject to periodical updating to incorporate modifications and improvements.

C Tutorial... A C tutorial is also handy if you want to understand more about C programming. There is a C Tutorial available in the Public Domain libraries. It has example C programs that are explained in the tutorial. These examples can be studied and tinkered with and compiled. The tutorial offers quite an easy comprehensive introduction to C programming. It costeth very little and is good value, so add it to your list of requirements.

C68, LibCport, C Tutorial and CfiX can be obtained from **David Walker 22 Kimptons Mead, Potters Bar, Herts EN6 3HZ, and from Qubbesoft P/D, 38 Brunwin Road, Rayne, Braintree, ESSEX CM7 5BU.**

ALL

Helen Armstrong nips over to Hammersmith for the latest All Formats Fair.

FORMATS

H missed two opportunities to make a London QL visit, I pounced on the chance to go to the London All Formats Fair on Saturday 14 November. Your best chance of meeting a QL trader at a Fair now is to visit the All Formats Fair at London or Glasgow.

Not far to go

There has been talk about the London fair being hard to get to since the organisers moved away from the long-established Horticultural Halls in Victoria. Ignore it! It's true that Sandown Park is much further out of town, inconvenient for non-drivers or visitors from the Continent (although there is a train service). But QL traders aren't all that interested in Sandown.

The London Novotel at Hammersmith is a different matter. It's out of the centre, but not by much. Typically 15-20 minutes on the Tube from (say) Charing Cross or Green Park will get you to Hammersmith. The service is flexible (there are three lines different parts of central London, and they all interlink) and reliable.

The Novotel is right by the Underground station. True, finding it is a challenge. There is a wall-map in the main entrance to the station (on the North side, away from the Hammersmith

Flyover), but if you have no memory for diagrams, simply walk out of the entrance, turn right along the pavement, and take the first underpass you meet, which will take you under the raging roundabout to its eastern edge. Hang a right along the foot of the concrete tower

blocks above you, walking towards the Flyover, and about half way along the block you will see an unpromising-looking slit in the sheer cliff-faces. You may see a sign that says "Common-wealth House" or something similar. Ignore it. Walk straight into the heart of the block till

you come to the coach park in about 50 yards, thread your way past the coaches, and on your right you will eventually see a set of low glass doors, hopefully with the All Formats Fair poster stuck to one of them. Go through the doors. You are there. It's a mere stone's throw from the station - much closer than the Horticultural Halls.

Once I reached the Novotel exhibition rooms, I was at home. There was a limited but pleasant bar that sold a very good cup of tea (fresh milk) and steamed sausage roll, and was also

doing sarnies, pasties, Perrier and coffee. I didn't see any alcoholic refreshment, but then, I wasn't looking for it. The Hotel itself (and several pubs) are a short walk away. The hall is windowless but light, the toilets civilised (no wellies needed), and there

were about 40 stalls in total, which was fewer than I expected.

**why do they go
on marketing
original versions
of their
programs, after
releasing SE
(Special Edition)
versions at a
higher price?**

Fort QL

In fact, it wasn't a busy day. I passed stands doing Atari business, and new-and-used

PC kit, as well as some user groups. None of them were busy. I didn't waste time on the rest of the show, but went straight to the QL enclosure.

I found Miracle Systems, TF Services, Bill Richardson (EEC) and (on behalf of Quanta) Ron Dunnett at their posts. Fred Toussi of Software87 showed his face. I was half-expecting to see Digital Precision there, but popular opinion was that Freddie is 'underground' (which usually means he's working on something) at the moment, and we didn't see him.

Stuart Honeyball of Miracle declared a 'first' for the show - "We're actually selling something", he said - by which he meant that the hardware on display (notably two Miracle ED drives) were actually exchangeable for hard cash. Normally, Miracle has a waiting list and takes orders. The catching-up is down to the phenomenal demand for Gold Cards, which has allowed Stuart to hire two assembly outworkers and stay abreast of Miracle's other product lines as well.

We now know that the Miracle graphics card will be boxed, with an I/O interface for a PC-style keyboard, and space for a Gold Card, disk drives and power supply. Miracle are still wondering whether the SCSI card they are working on will be part of the package, or separate. Listening to the conversation, I would put a bob on it being separate. All the software is being written by Tony Tebb, but there are no announcements about specifications or names yet. And the time scale?

Miracle are cautious as usual. The graphics card is "Not far off," says Stuart. The PC plug-in card will be ready "before too long". As for the SCSI card, "Mike's just about done that. He's just got to draw something on a piece of paper." Everyone's having a laugh by now. But Miracle master the vicissitudes of scheduling with a policy of not advertising till the

JOCHEN MERZ SOFTWARE

Im stillen Winkel 12 — D-4100 Duisburg 11
Germany — Telephone & Fax: 0203 501274

NEW! Mailbox:
0203-591706
(8N1-9600-24h)

Best Wishes for 1993 — the year starts with good news: all prices listed here include 15% German VAT now. For goods delivered to other EEC countries you will not be charged any other tax, not even your VAT (which is definitely higher than ours!) For orders from outside the EEC, deduct 15% because you will be charged the VAT rate of your country (as before). All prices in DM now — which means that credit card purchases will be in DM and other £-payment has to reflect the exchange rate at the time of purchase.

QVME — The Breakthrough in Display Handling! You can now run QDOS on Mega STE's! The card plugs into the VME bus. No soldering, fully programmable screen resolution which can even be changed while the system is running! If used with a good Multisync monitor, you can get resolutions up to 1024x800 pixels and more!!! Of course, you can get the good old 512x256. We are also working on the TT software. With an adaptor for **DM 139,-** you can plug it into Mega STs. The price is **DM 809,-** write for more details!

QD 4 — Editor using the Pointer Environment, with many new options: GOTO BASIC Procedure/Function, Label, improved parameter string, print, search/replace, and MenuConfig (the same as Config, but menu-driven). **DM 122.50** Upgrade from V3 **DM 31,-**

QSpread — Spreadsheet for the Pointer Env. More than 32000 cells. Completely pointer-driven. Formulae-orientated, 22 scientific functions, macro functions, help, scrap, etc. [V1.09] **Largely improved, with better manual DM 169,-**

QMenu — Menu Extension V3 QMenu is an interface to pre-defined menus (e.g. file-select, simple-choice boxes, select from lists) which are very easy to use. These menus may be used by SuperBASIC, machine code and other languages. The menus are very comfortable, multi-column etc. **DM 39.90** Update from previous version with new manual **DM 16,-**

FIFI — the FileFinder FIFI is extremely useful and saves a lot of time: it scans devices or directory trees and searches fields or filenames for strings. Combinations are possible, e.g. String1 AND string2 BUT NOT string3 etc. **DM 49.90**

QDOS Reference Manual — This book is a must for all machine-code programmers. It explains how to use QDOS, all traps and vectors, the Thing System, the HOTKEY Syst.II and much more. It shows which features work on a QL, an Emulator, how to write compatible for future operating systems. 170 pages. **DM 89.90**

QPTR — The Pointer Environment Toolkit Revised manual which describes the use of the Pointer Interface and the Window Manager from SuperBASIC and machine code. Examples on disc, keys, macros & extensions for BASIC. **DM 92,-**

FLP/AM Level 2 with ATR!!! Replacement EPROM for QBoard (V1.17 onwards, with & without mouse) or TrumpCard (please specify!) Real sub-directories! About twice as fast, better slaveblock-handling! (not on JM-ROMs) **DM 56,-**

EASYPTR — Create your own Pointer-Env. menus & sprites, use them in your SuperBASIC or machine-code programs. Supports all the PE facilities, even split application sub-windows and uses the Menu Extension! Many examples!

Part 1 Basic with menu generator, sprite editor etc **DM 99,-**

Part 2 SuperBASIC Toolkit with Appendix manager & many examples **DM 49,-**

Part 3 Library routines for C and Assembler and Source code generator **DM 49,-**

SYSTEM — System Tools for QDOS: real file-attributes, e.g. write-only, hidden or for selected users only. Works even over net and on winchester. New TRAP#3. New devices: named pipes, MEM, NUL > 80 new procs & features. **DM 90,-**

QDesign II — more features, improved and more printer drivers (incl. laser), ASnap which scans text from screen and puts it into scrap, virtual drawing mode on Emulator and Minerva etc. **DM 150,-** Upgrade from version 1 **DM 46,-**

VecEdit Vector — Font-Editor for QDesign DM 60,-

SER Mouse V2 — software driver which lets you connect a serial (e.g. IBM-style) mouse to one of the SER ports of the QL, which mimics the QIMI. **DM 40,-**

The Lonely Joker — three great patience card games. Needs Toolkit II **DM 52.90**

Diamonds DM 35.90 — BrainSmasher DM 45.90 — Arcanoid DM 35.90 —

Firebirds DM 35.90 — SuperGamesPack DM 90,- — QShang DM 45.90 —

QSUP System Utility Pack DM 79.90 — Thing & Eprom Manger DM 61.50

DISA (Intelligent Disassembler DM 90,-

NEW: E-Driver software (for all QL-Emulators) with new initialisation: boots faster, runs faster, needs less memory, boots from harddisk without EPROM, more features inbuilt etc. The way forward, as level D ends at D.16. With new manual **DM 49.90**

Technical Printer Manuals:

EPSON ESC-P/2 for all EPSON 9/24/48 pin printers. 500 pages **DM 90,-**

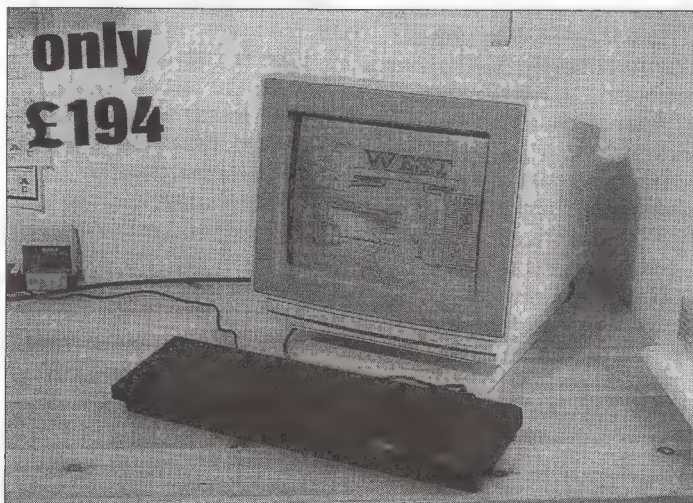
HP Deskjet for the Deskjet family (including 500C) **DM 56,-**

Please add DM 13,- for p&p (Europe) or DM 13,- for one item and DM 7,- for every further item (Overseas). All prices incl. V.A.T. (15%) E&OE.



Please write or call (English & German spoken) for more information.

FOLLOWING OUR QL OFFER WE CAN NOW SHOW YOU A CONSIDERABLE SAVING ON A SUPERB QL DEDICATED COLOUR MONITOR



GOLDSTAR EGA MONITOR

PRICE ONLY
£194
INC. VAT WITH
BASE ADVERTISED
PRICE £204 RRP
OVER £300

- 14" Screen
- 85 Column
- Mains & data cables
- Tilt & swivel base
- Smart, modern styling
- 0.31 Dot Pitch

PACKAGE OPTION WITH MONITOR, IF YOU ALSO BUY PRINTER OR DISK DRIVE SHOWN IN WN RICHARDSON & CO ADVERT ON PAGE 46. A FURTHER £10 CAN BE DEDUCTED E.G. MONITOR PLUS SAMSUNG PRINTER FOR £319 WHICH IS LESS THAN THE NORMAL RRP OF EACH ITEM

ORDERS SHOULD IN THIS INSTANCE BE PLACED
DIRECT WITH WN RICHARDSON & CO. AS PER
THEIR 1/4 PAGE ADVERT ON PAGE 46

THE NEW USER GUIDE

KEYWORD INDEX

SECTION
TWENTY
ONE

This month in the Keyword Index, Mike Lloyd starts with ON and finishes, appropriately for the festive season, with a PARTYP.

ON x GOSUB line1,
line2, line3, line4 ...
ON x GOTO line1,
line2, line3, line4 ...

LOW-LEVEL PROGRAM STRUCTURES

x A numeric variable
line1, line2, etc. Valid line numbers

ON...GOSUB has honourable antecedents back through early Basic dialects to second generation languages such as Fortran. The idea is that the interpreter could be sent to various parts of a program depending upon the value of a variable. The programmer identifies the first line number of each subroutine in a comma-separated list. Prior to reaching the ON... GOTO or ON... GOSUB line a numeric variable is manipulated to point to one of the numbers in the list (ie if X equals 4, the fourth line number is selected from the list). Whether the interpreter returns from the jump depends upon whether GOTO or GOSUB is used (refer to these keywords in an earlier *New User Guide* for guidance). Even though SuperBasic avoids the obvious trap by updating the list automatically when the program is renumbered, these commands remain awkward ways of branching a program and are best avoided. It is generally preferable to divide programs into procedures and functions and forget about GOTOS and GOSUBs in all their variations.

OPEN #chan, name
OPEN_IN #chan, name
OPEN_NEW #chan, name
OPEN_OVER #chan,
name
[Super Toolkit 2]

DEVICE HANDLING COMMANDS

#chan A valid channel number
name A filename, console definition, screen definition, printer definition or network definition.

Most Basics are interested only in opening files, which would make the OPEN family simple file handling commands. Qdos brings together printers, file storage devices, networks and screen windows with the unifying concepts of "devices" and "channels". Imagine a channel as being one of a line of tubes. The far end of each tube can be attached to a screen, printing, file storage or network

device. Once a tube is connected the size, shape and type of the device at the other end are of little interest; the attributes of the tube itself are much more important. Some tubes allow one-way traffic (typically when attached to printers or files), while others permit two-way communication (which applies to consoles but not to screens).

The OPEN command tries to implement two-way communication between the QL and the device, although the attributes of the device may limit this. OPEN_IN is used only for opening a file in "read only" mode. This is a safety measure to prevent accidental file corruption by incorrectly trying to PRINT to it. The OPEN_NEW variant is also used only with file storage devices. It opens a new file with the given name before channeling output to it. OPEN_OVER will attempt to open a new file with the given name, but if a file with that name already exists it will be replaced and overwritten by the new one. There are many errors which can be generated when assigning channels to devices, but SuperBasic's answer is simply to halt processing with an error message. *Super Toolkit 2* is equipped with a set of functions such as FTEST() and FOPEN() which test to see if OPEN is possible without bringing a program to a halt. *Turbo Toolkit* provides the same facilities with DEVICE_STATUS().

Once a channel has been connected to a file output is passed along it using PRINT#chan. Input is obtained from the channel using INPUT#chan or the INKEY\$#chan() function. In this way information can be passed to a screen, the network, a printer and a file using the same set of commands. A simple procedure to echo everything passed to a screen to the printer (or to a file) is:

```
100 OPEN#5, ser1 : REMark connect channel 5 to the printer
110 ECHO "Here is a line of text"
120 DEFine PROCedure echo (text)
130 PRINT text
140 PRINT #5, text
150 END DEFine echo
```

Note that channels #0, #1 and #2 are already open to screen consoles when the QL is booted up and that channel #0 allows SuperBasic commands to be typed in. If channel #0 is reassigned its special characteristics are lost until the QL is rebooted. In Turbo programs channel #0 is not recognised by the interpreter as a source of direct commands.

The syntax for non-file devices will be dealt with in the Concepts section of the New User Guide.

OPEN_DIR #chan, name [Super Toolkit 2]

FILE HANDLING COMMAND

#chan	A channel opened to a file storage device (eg a microdrive)
name	The name of a device (eg "flp1_")

One of the earliest gripes about SuperBasic was that it was difficult for a program to read and manipulate the files on a microdrive cartridge. The only method available seemed to be to open a file and write a directory listing to it (eg OPEN_NEW#3, flp1_dirlist: DIR#3: CLOSE #3). Once populated and closed, the file could be re-opened in read-only mode and each line read with an INPUT statement. Super Toolkit 2 came to the rescue with OPEN_DIR, but the documentation made scant reference to it and failed to demonstrate how it could be used.

The OPEN_DIR command actually opens a read-only channel directly to the file map at the start of every microdrive cartridge and floppy disk (if floppy disks can be said to have a start). Carriage return codes only occur by accident in the file map so characters must be read one by one with the INKEY\$() function. Fortunately there are a fixed number of bytes per file entry, so a simple loop will gather discrete information about each file. The first 16 bytes of the map refer to the medium itself and can be skipped over. Thereafter each block of 64 bytes refers to a given file. The following routine quickly prints the bytes to the screen. The full decode lies beyond this Guide, but the file name will be obvious at the start of each line.

```
100 CLOSE #6: REMark only if it is already open...
110 OPEN_DIR #6, FLP1_
120 FOR x = 1 TO 16: junk$ = INKEY$(#6)
130 REPEAT loop
140 entry$ = ""
150 FOR x = 1 TO 64: entry$ = entry$ & INKEY$(#6)
160 PRINT entry$
170 IF EOF(#6): EXIT loop
180 END REPEAT loop
```

OPTION_CMD\$() [Turbo Toolkit]

COMPILED PROGRAM PARAMETER FUNCTION

The Turbo compiler provides many ways in which tasks can pass information to each other. One of

the simplest is `OPTION_CMD$()`. Programmers can call any compiled program using the `EXECUTE` family of commands and add after the program name a string containing anything which might be useful. This string can be read within the compiled program using the `OPTION_CMD$()` function. The calling line might be:

```
EXECUTE flp1_task "mdv1_workfile, backup, 7"
```

Within the `flp1_task` code there might be written a function to strip away the text separated by commas to assign values to three variables. The function could be called with the line:

```
140 filename = DECODE(OPTION_CMD$, 1)
150 process = DECODE(OPTION_CMD$, 2)
160 refnum = DECODE(OPTION_CMD$, 3)
```

Remember, what you place in the string passed to the executable program and what you do with it within the executable is up to you.

OVER #chan, value

SCREEN ATTRIBUTE COMMAND

#chan	(Optional) A channel linked to a screen window or console.
value	Minus one, nought or one.

The `OVER` command plays clever games with the QL's screen map to produce a limited variety of special effects. `OVER 0` is the default and everything is printed in the expected `INK` and `PAPER` colours. `OVER 1` has no effect on `INK` colours but makes the `PAPER` colour "invisible", allowing existing backgrounds to be seen directly beneath the text being printed. This can have the effect of making text difficult or impossible to see. If you have a red background and set `INK` to red and `PAPER` to black, text will normally be clearly visible. Following an `OVER 1` command, though, the black background will not be printed, so any further `PRINT` statements will put red ink on the existing red background. `OVER 1` has its uses when mixing text and graphics because lines beneath text are not obscured by the strip.

`OVER -1` is a different effect altogether. Once `OVER -1` has been issued any text or graphics printed in the affected window take their colour from a combination of the existing background colour and the current `INK` colour. Should you need to know, this is done by a bitwise Exclusive OR. The non-technical need bear in mind only that `OVER -1` is a way of ensuring that no matter what background colours are in use that text or graphics placed over them will adjust their colour so that they remain visible. Different degrees of contrast can be achieved by selecting different `INK` colours, with white (`INK 7`) providing the most visible contrasts and black (`INK 0`) being entirely invisible.

PAN #chan, pixels, area

SCREEN MANIPULATION COMMAND

#chan	(Optional) A valid screen channel
pixels	An integer representing the number of pixels to move (can be negative)
area	(Optional) The part of the screen to pan (3 = cursor line, 4 = cursor line to right of cursor)

Panning is what film directors do when they want the camera to sweep across a scene, and the SuperBasic `PAN` command can be used to give that illusion. What `PAN` actually does is to move screen pixels horizontally, replacing those which "fall off" the screen on one side with `PAPER`-coloured pixels on the opposite side. To emulate camera panning the `PAN` command has to be called in a loop. The panning effect can be limited by an optional third parameter to the cursor line (with a value of 3) or just that part of the cursor line which lies under or to the right of the cursor (with a value of 4).

It would appear that `PAN` and `SCROLL` share a single parameter set in that `SCROLL`'s optional screen areas are 1 (above the cursor line) and 2 (below the cursor line). If it was ever intended that `PAN` and `SCROLL` would take any of the four values the QL's ROM was never developed in that direction. `SCROLL`ing the cursor line vertically and `PAN`ning the top and bottom parts of a screen window seem to be excellent ideas which programmers unfortunately have to implement for themselves.

The quickest way of moving part of a window is to locate another window over the top of it and panning that. The following code gives an example:

```
100 MODE 4: WINDOW 512, 256, 0, 0 : PAPER 0 : CLS
110 FOR x = 1 TO 200: PRINT FILL$(CHR$(32 + RND(100))), 5); " ";
120 FOR x = 1 TO 100: minipan 300, 20,80, 40,2
130 DEFine PROCedure minipan (wide, high, xpos, ypos, pixels)
140 OPEN#8, scr_: WINDOW#8, wide, high, xpos, ypos
150 PAN#8, pixels
160 CLOSE#8
```


170 END DEFine minipan

**PAPER #chan, main,
sub, stipple**

SCREEN HANDLING COMMAND

#chan	(Optional) A screen channel number
main	An integer between 0 and 7 (for solid colour) or between 0 and 255
sub	(Optional) An integer between 0 and 7
stipple	(Optional) An integer between 0 and 3

The PAPER command sets the background colour for the screen and affects the colour underlying printed characters and the screen colour used by CLS. The complete range of colours, including all the stipple combinations, can be represented by a single value between 0 and 255 that PAPER can take as a parameter. Alternatively, colours can be built up from three parameters, the first two determining the colour combination and the third the stipple. Thus a red and cyan checkerboard stipple is obtained by PAPER 2, 5, 3.

**PARNAM\$(param)
[Super Toolkit 2]**

PARAMETER HANDLING FUNCTION

param	The location of a parameter in a list of parameters.
-------	--

In SuperBasic, parameter names take the place of the true names of values. This means that any change to the value represented by a parameter within a procedure will survive after the procedure call is complete. There are times when it might be useful to know exactly which variable has been passed as a parameter to a procedure. We might have a labelling procedure for fields in a database, for example, which prefixes a title to the field it is printing. PARNAM\$ would do this task:

```
100 surname$ = "SMITH": forename$ = "JANE"
110 LABEL surname$: LABEL forename$
500 DEFine PROCEDURE LABEL (field$)
510 LOCAL result$
520 IF PARNAM$(1) = "surname$": result$ = "Surname is "
530 IF PARNAM$(1) = "forename$": result$ = "Forename is "
540 PRINT result$ & field$
550 END DEFine LABEL
```

**PARSTR\$(varname,
paranumber)
[Super Toolkit 2]**

PARAMETER HANDLING FUNCTION

varname	the "local" name given to a procedure's parameter
paranumber	the location of the parameter in the parameter list

Even Tony Tebby describes the PARSTR\$ function as untidy. It is certainly able to confuse. Its two arguments are a parameter name and its location in the parameter list. It returns either the "true" name of the variable represented by the parameter or, if that variable is a string, the value of the string. It must have been useful for something.

**PARTYP(varname)
[Super Toolkit 2]**

PARAMETER HANDLING FUNCTION

varname	the "local" name given to a procedure's parameter
---------	---

Another way of finding out more information about a parameter is provided by the PARTYP() function. When given a parameter name it responds with a value of 0 if the parameter is null, 1 if the parameter represents a string, 2 if it represents a floating point number and 3 if it represents an integer. This is particularly useful because parameter names do not have to follow the same variable naming rules elsewhere in SuperBasic: a parameter without a dollar sign suffix can represent a string. Indeed, because of SuperBasic's type coercion a parameter could represent a string in one procedure call and an integer in the next.

Hundreds of new programs are sent to software distributors by hopeful authors every year. Software distributor Richard Alexander provides a few (!) tips on the best approach.

So you want to be a SOFTWARE AUTHOR?

I've written the notes in this article to try and remove some of the potential problems that software publishers face when they get programs for evaluation, and to help authors get their software accepted. This article draws on my experiences as a software publisher and a Public Domain Library organiser. I'd also like to thank Dilwyn Jones for his comments on the first draft and for allowing me to use a few of his programming hints. I hope that authors will find this of use to them - comments and suggestions for improvement from the "other side of the fence" are most welcome.

First of all

Before you even start writing the code - check what is already available. There is comparatively little software currently on sale for the QL, and the market is hardly increasing, so don't waste time writing a program that simply does what an existing one does, unless you're submitting it to the Public Domain or your program is clearly to be a very obvious improvement. Also, if the program is intended for commercial release it's wise to check with prospective publishers that there's a market for it (even if you

eventually intend marketing it yourself, a little market research can save a lot of wasted effort).

You can also find out if someone else is planning to do a similar program - a joint venture might be possible. As for which publishers to approach, it's a good idea to look at what else they sell. Also, find out whether all their programs are developed in-house or they sell other people's software. You can try and contact authors who are published by them - to see how often they get paid royalties (if at all), what their attitude to authors and the public is, and whether they are likely to be still in business when your masterpiece is finished.

If you are just beginning to program for the QL (and fortunately there are such people) it might be an idea to submit your early efforts to the Quanta Library, or to the established Public Domain Libraries, so they can let you know whether your style of presentation and programming are of a suitable standard for commercial release. Have a close look at other people's programs, see if you can work out what features you like about them (such as ease of use or clear uncluttered screens) and which you don't (lack of error-trapping, bad spelling or grammar, incompatibility with other programs, etc.). Try to avoid repeating other peoples' mistakes.

Commissions

A few QL programs are still written to satisfy the demands of publishers - they require a particular type of program and contract programmers to undertake the work. If you are doing a program on this basis make sure you have a written contract before commencing work. Also make sure that you get a legal adviser to check through it. Normally these contracts assign the copyright to the publishers, so I'd advise that you ensure that the copyright returns to you if the publisher fails to pay you, or goes bust, or decides not to publish the program. This way you'll be able to take the program to another publisher or release it yourself.

In fact, I'd recommend this to all programmers even if your programs are sold on a royalty basis. A number of good programs have dropped out of circulation because the copyright holders cannot be determined or traced. Another point to bear in mind is to ensure that any unsolicited programs sent to a publisher are done so "subject to contract", that is unless the publisher makes a formal agreement to sell your program, then they are obliged to return your program and destroy any copies they may have made. This should make it easier for you to sue them should there be any disputes over copyright.

Planning

A lot of programmers plan the program as they write it; others prefer to write a specification before a line of code is written. I would recommend as good practice that you write down as detailed a spec as possible, and that you update this as the program progresses. If necessary, keep a folder with your working notes in one place so you can see what you intend doing and what still needs to be done. If you are really systematic in your working, it is a good idea to make a record of all the line numbers of the program (if programming in SuperBasic) which you change between versions. How you actually do your coding is, however up to you - what language or development tools, etc., you use. The important thing is that you know what your code is supposed to do! A surprising number of programmers don't, with resulting errors and delays.

One important point - if you are intending to compile SuperBasic programs, get the latest release of the compiler, as these will have been upgraded to take the latest hardware improvements into account. Similarly, if you're using the C68 Compiler, get the latest releases as they come out and be prepared to recompile programs (or modules thereof) as

necessary.

Whichever language you use, it is most advisable to ensure that the program can be started using the EXEC command as opposed to EXEC_W, that is whenever an INKEY\$ statement is used, you'll need to have enabled a cursor. With so many programs multi-tasking, Ctrl-C'ing into a program which has no cursor enabled effectively locks the user out. EXEC_W'ing a program also does not allow you to continue with any other SuperBasic commands until its job has finished.

Similarly, as most multi-tasking programs use up a fair bit of the screen, and not everyone will be using a task and display manager, it is wise to include a screen redraw key (normally F4 or Shift-F5) to redraw menus so that users know what options are open to them when they return to a program.

Environment

Very early in the development cycle, you'll need to decide what environment you want the program to run in. Will it be Mode independent, will it need a monitor or can people with televisions also use it? What media does it need - can users run the program from disk, mdv, ramdisk or hard disk? Even these days there are still many QL users without monitors or disk drives whereas others are continually upgrading their hardware to the latest specification. If at all possible, write your software so that as many people as possible can use it. Don't assume that just because it runs on your set-up it'll run on all (or any) of the others.

Do try to avoid incorporating commands or features that may be integral to your own set-up but not other peoples'. I'd recommend, for example, not using commands from *Toolkit 2*, *Minerva*, or those found on *Trump* or *Gold Cards*, especially if these are going to be in compiled programs as they will prevent users even loading them. The exception is that it is fairly - only fairly - safe to assume that disk drive owners will

have *Toolkit 2*. On the other hand if your compiler has a toolkit that can be distributed, by all means include the commands - or use commands such as those found in *DIY Toolkit* or *PD toolkits*. Whichever toolkits or commands you use, keep within the copyright restrictions and send the copyright owners a complementary copy of the program!

Regarding memory use - keep it as low as possible, or be prepared to make two versions available, one for standard QLs and one for expanded versions. Similarly, one can produce versions that fit on mdvs and enhanced ones for disk users. However, the more versions you produce, the more work you are letting yourself and your publisher in for, especially if you have to debug, re-write and recompile each version every time a problem is found.

Tony's Pointer

Increasingly, especially in Europe, programs are being written that can operate under the Pointer Environment - it may be worth the effort to make your programs compatible (at least) with the Pointer Environment so that users can safely multitask the programs. This is especially important if you are writing utility or applications software as these may well be required to multitask. If, however, you are writing games aimed at the QL user with only a standard 128K machine, this is less important. If you require further information on this then Jochen Merz can supply the QPTR (Pointer Environment Toolkit) which includes a revised manual. Also available from him is the *Qdos Reference Manual* which shows how to use Qdos, etc. More importantly, it explains how to write code that is relatively future-proof and which can run on a variety of Qdos platforms.

With reference to the Pointer Environment, but good practice in any case, you should not resize the area of the screen used by your program. If you need to change the size or location of

windows used by your program, open a screen window covering the the largest area used by the program to guard the program's display from causing problems when running under the Pointer Environment. However, you should not define a window area that is over-large, as the Pointer Environment does not generally allow two jobs with overlapping windows to write to the screen at the same time.

Does it work?

One of the main problems that authors face is getting their programs to run properly. Equip yourself with a couple of reliable de-bugging tools so you can at least track down where the problems occur. Never send in programs that do not work on your own system in the hope that either the publisher will not notice them or that they will be able to fix them for you. Most software publishers will happily look at programs that are still being developed but few have the resources or time to do your de-bugging for you. If you want to show a publisher what the program will be able to do, once you finish it, send a demo version - a game with just a few screens, or a simplified version of a utility or application, but make sure that what you send works. If you know there are bugs or features that you need to sort out, point these out in the letter that accompanies the program, together with a description of what you intend to do about it.

There are also a few general housekeeping rules that you should observe to make your life easier (and make things easier for potential publishers - which always goes down well). Do backups of the program at regular intervals and give each one a slightly different name. It's often a good idea to use an incrementing filename, such as "myprog12", "myprog13", etc. It's a good thing to use filenames that clearly indicate what the program is. (I wish more article authors would do this - Ed.) Use the conventional file suffixes, such

as ".cde", ".asm", ".bas" so you can see at a glance which are your source files and which are the compiled or assembled ones.

File-naming

Once the program is complete, if your program is destined for the Public Domain it is advisable to give the files the same name and then use the suffixes above to differentiate the types of files. This way they can be put on a disk with several other programs or be easily renamed if they clash with existing filenames. However, I'd advise against using hard directories on any disks you submit, as the program is unlikely to be eventually distributed in that form. If your system allows it, make sure each file is correctly date-stamped. And don't put all your programs and backups on the same disk. Use two or three, so that if one goes down there's a fairly recent version you can work from. (This also underlines the need for keeping notes on the program, so you know what you've done between versions.) Do label the disks you're using clearly.

Boots, boots, boots

Before you even think of sending your masterpiece for evaluation there are a few more things you should do. Firstly get someone who isn't familiar with the program to check it through. You'll know what it's supposed to do and how it's supposed to do it, but neither the publisher or a first-time user will and initial impressions are very important. This test should start with the computer switched off, and finish with quitting the program in good order. It's amazing how many people send in programs which have no indication how they're supposed to be booted.

At the very least write a "boot" program that sets (if necessary) the mode and links in any toolkits, extra fonts, etc., and then launches the main program. It may not be necessary to set the mode, as you can check to see what

mode the program is in by using the DIY Toolkit extension SYSBASE. To check, for example if the machine is in Mode4, you use the line "IF (PEEK(SYSBASE+52)&&8)=8 THEN MODE 4". Remember also that users with AH and JM QLs cannot run programs if they need extensions loaded by that program. Better to load the extensions in a short Boot program, and then EXEC or LRUN the program.

It has been suggested that users of *QLiberator* should link the extensions and runtimes to the program but this can result in a waste of memory if there is more than one *QLiberator* program running at the same time. At the very end, please leave QL in the state you found it in initially; don't have odd windows or channels open or the main ones (#0,#1 and #2) inaccessible.

Use REMs

Other useful hints on boot programs: don't use un-numbered boots. These are very difficult to amend, both for the publisher and the end user. Use REM statements to allow options such as loading *Lightning* extensions. It is probably easier to have separate disk and mdv boot programs, most publishers keep separate "master" copies for programs to be run on different media, but do make the boot program configurable by users, so that they can easily run the program from either set-up. Similarly if the program requires a randisk, REM it so that people with hardware ramdisks don't find themselves loading additional ramdisk code.

If the program is designed to multi-task, it is a good idea to provide a Boot program that can be merged with an existing boot to enable the user to use it easily with any multi-tasking environment. Finally, please leave space in the boot program to allow both the publisher and/or the user to add further lines. (For an example of a well-written Boot program, see the sample boot program written by Rich Mellor and Alan Pemberton for their *SToQL* program.)

Another very useful addition

Example listing for the Submitting to Software article.

```
100 REMark ST-QL SCREEN CONVERSION PROGRAM - Version 1.24
110 REMark ?1991 Alan Pemberton & Rich Mellor
120 REMark Released by CGH Services, Cwm Gwen Hall, Pencader, Dyfed, Cymru
SA39 9HA
130 :
140 REMark If you have lightning or speedscreen, then delete the REMark from
the appropriate line
150 REMark _LNGinit
160 REMark a=RESPR(22360):LBYTES FLP1_LNG_TEXT_EXT,a:CALL a
170 REMark a=RESPR(4316):LBYTES FLP1_LNG_GRAF_EXT,a:CALL a
180 REMark a=RESPR(6916):LBYTES FLP1_LNG_MATH_EXT,a:CALL a
190 REMark _SPEED 1
200 :
210 REMark If you have Toolkit II, then delete the REMark from line 210
220 REMark TK2_EXT
230 :
240 REMark If you have the new ATR device drivers, delete the REMark from
line 250
250 REMark ATR_DEV
260 :
270 WINDOW 512,256,0,0:PAPER 0:MODE 4
280 INK #0,0
290 a=RESPR(344):LBYTES flp1_edline_ext,a:CALL a
300 a=RESPR(2680):LBYTES flp1_CGH_ext,a:CALL a
310 a=RESPR(1384):LBYTES flp1_fn_ext,a:CALL a
320 a=RESPR(270):LBYTES flp1_pixel_ext,a:CALL a
330 a=RESPR(10600):LBYTES flp1_qlib_run,a:CALL a
340 a=RESPR(144):LBYTES flp1_header_ext,a:CALL a
350 a=RESPR(204):LBYTES flp1_blank_ext,a:CALL a
360 INK #0,7:CLS:WINDOW 448,200,32,16
365 :
370 REMark If you want to run SToQL on Minerva's second screen
380 REMark then ensure Minerva is in dual screen mode (press F4 on
390 REMark start-up) and remove the REMark from line 400
400 REMark MODE 64+32,-1
450 :
999 REMark Do not renumber past this line!
1000 EXEC_W flp1_convert_obj
```

to a working version of the program is a "clone" program. This will enable the user (not the publisher, who will probably make copies using a sector copier for speed) to make authorised back-ups onto whichever media they want to run the program from. At the very least the clone program should allow users to change easily the source drive, the drive the program is being copied to, the drive it will run from when cloned and, if necessary, the drive on which any data is to be found.

Write, now

Even if your program has extremely clear on-screen prompts, it will inevitably need some documentation supplied with it to help the user and to stake your claim to copyright. Whereas the publisher will normally be responsible for the eventual publication of the manual, you'll make yourself much more popular if you can at least write enough of the instructions in clear language so that the publisher can work out what the program is

supposed to do and how to go about achieving it. Please put this through a thorough spell-checker before submitting it and, if at all possible, get someone else to read it through. While publishers can cope with hand-written documentation, your program will be dealt with more quickly if the documentation is supplied both as a text file on disk and as hard copy. Personally I'd recommend sending Quill "_doc" files, as everyone still has Quill, unless your publisher suggests an alternative that you are both happy with. And while you're at it - double-check the spelling of any text that appears on-screen as well.

A nice touch, but not essential at this stage, is to do an attractive title screen. It may not make much difference to the use of the program but it can help get people interested at shows and so forth. One proviso: saving a full QL screen (uncompressed) on a mdv will eat up a lot of space, it may be better to do a simpler one that the QL can draw quickly.

Check, check, re-check

Once you've tidied up the program and the presentation and had yet one more look at it, do try and get another, trustworthy, QL user to have a look at it. More than likely they will have a different set-up to yours (different rom, memory, etc.) and this may overturn any assumptions you've made about what hardware the program is going to run on. They should be encouraged to try booting the program up using both F1 and F2 in both TV and Monitor modes. If problems are encountered, do try to sort them out before sending the program in. A program that only runs on your own machine, or others of an identical specification, will rarely be accepted for publication - and even Public Domain Libraries have been known to chuck out programs which are fussy about the hardware they're running on.

Finally, before you part company with a COPY of your program make sure that what

you are sending is the latest version. Clearly label the disk with the program name and version number and your name and address. Then write or phone the target distributor or publisher and check that they are still interested in seeing the program.

It's also advisable at this stage to check what your likely earnings are going to be. It may pay to shop around. A legal contract is probably not necessary at this stage. Then deposit a copy with your solicitor (just ask them to keep it in a safe with a note of the date, and sign a declaration that the enclosed disk contains a program which is all your own work.) (Heaven help you if it isn't!) Now you can send your program off.

Time factors

The first thing that authors must realise is that publishers are rarely dealing with just one program at a time. Many have half a dozen in development at any one time, and they are also responsible for the backup on all their other products. As many of these will also be in the process of upgrading (to take account of new roms, compiler versions, etc.) and they'll have their sales to attend to (and in many cases a full time job) you'll soon realise that your new program, which may take some time to get into, is not necessarily going to be put on the top of the pile for attention. At best they will take a quick look at it, see if it loads, try it on two or three QL systems and then put it to one side for a more detailed look later.

Some publishers may also have informal agreements for specialist programmers or users to look over programs rather than do it all themselves. So you'll need to wait a couple of weeks at least before expecting a reply. It is probably a good idea to give the publisher a reasonable deadline for replying to you, say four weeks. If after that length of time you haven't heard from them, or they won't commit themselves to publishing the program, then write saying that you are going to offer the program to another publisher (assuming you can

find someone who will take it!) and demand that the original prospective publisher either reply by return or undertake to destroy the version you have sent to them and not distribute it or any copies thereafter unless it is with your agreement.

So: what publishers are looking for are programs that work first time on all available QL systems; that have no awkward system requirements; and which are bug free. Sometimes we get these! Other programs, however, require far more work before they can be considered for release and I'll go through some of these in more detail.

Printers again ...

One of the main stumbling blocks is printers. Few publishers will have access to more than two printers and, as we all know, no two printers work in exactly the same way. So if you submit a program designed to print its output, be prepared for plenty of problems. Luckily Joe Haftke has very kindly placed into the QUANTA library a collection of printer control codes, so I'd advise all authors working on programs which can have printed output to get a copy of this. It would certainly be a good idea to include two or three different printer control files with your program when you submit it. Another idea would be to use the GPRINT prt file used with Abacus, as users may well have adapted that to suit their own system.

As a rule, I'd advise against embedding too much printer control in the main program. Have it as a separate module, making de-bugging and amendments to the program easier. Another useful option is to have a save-screen function so that users can output the screen file from a specialist screen dump program, or import it into a graphics package to add titles, etc. (But don't assume that everyone has the Trump and Gold Card Screen Dump code available to them!)

For games ...

Publishers will attempt to test the program as far as they can, and anything that authors can do to help this process is most welcome. For example, if you are submitting an adventure, enclose the solution and a map so that the testers can play it right through. Nobody wants to release a program that people can't complete, and the more you can do to assure publishers that your program can be completed, the quicker and more likely they are to take it on board. A solution will also help in the future when players get stuck and require help with the program!

In practice I have found that very few games require much alteration once they have been sent in. They either work first time, or they don't, as the case may be. Provided the publisher can get past the first few screens they'll probably take your word for it that the rest are achievable - but the inclusion of a cheat mode (with infinite lives or whatever) makes the process a lot easier! This can be left in the release version as there are people who actually get their enjoyment from finding such cheat modes and others will find it useful if they get totally stuck on the game. However, do try and prevent this information getting to the punters too soon after release - it spoils their fun.

... and Utilities

Regarding more serious applications, if your program relates to hard disks or particular hardware or software set-ups, it is possible that the publishers won't have that set-up and will need to send the program off to another QL owner to test it. This will inevitably delay any response to the initial sending-in of the program, so check what the publisher intends doing with the program before sending it in. If you have Trump Card version 2, Gold Card, the Level 2 drivers, or any other system with hard directories you can test hard disk operation by creating a directory on disk and then using "WIN_USE flp" to simulate running from hard

disks. (My thanks to Dilwyn for this tip.)

Even if the program you send is complete, it is quite possible that publishers will suggest improvements. These can take the form of altering the user interface to make it tidier or perhaps to incorporate mouse control. It is also possible that a publisher will have established a house style, or have available a front-end system that can be bolted onto your program. Don't be offended by suggestions of this nature. If it makes your program quicker to run or easier to use then it's probably better to go along with what the publisher says. Here you'll have to decide whether you want the publisher to have a copy of your source code (so that other programmers can amend it) or whether you are going to hang onto it to prevent other people pinching your code. (Once you've sent in your program it's virtually impossible to prevent people pinching your ideas!)

Features flourish

Another area for improvement that publishers look at is the range of features. With applications, increasing features can be a never-ending task, as there's always something else that you can add to a program. Note that, contrary to some cynical views expressed in the computer press, this is not a case of publishers releasing incomplete programs, but rather responding to ideas from users to improve already existing programs. Publishers also need to keep up with the competition - if your program can be compared with a similar one, then they'll be asking for upgrades to keep your program competitive.

Enhancements in the QL hardware can also be an impetus for improving programs. The release of the Gold Card with its 2MB ram and much faster processor means that memory and speed limitations on applications can be removed, so authors may need to keep the options open on a program or make them user-definable, rather than

assuming a fixed memory space or processor speed. Further, the possibility of the QL being run on "alien" platforms, such as the existing ST (hardware) and Amiga (software) emulators and the forthcoming PC emulator (not forgetting the option of using the Thor's Mode12) means that additional screen modes may need to be catered for to take full advantage of the improved hardware environments. And, who knows, we may eventually have the option of decent sound on the QL as well! (Another point about some Thors: sound should be an option that the user can turn off. The THOR handles sound differently from the QL and it can cause interminable delays if it encounters sound parameters it can't handle properly.)

Future-proofing programs is virtually impossible, especially when authors are still writing programs on AH QLs without memory expansion. Again, publishers may suggest that a modular construction to the program which allows for selective upgrading of parts of it, while leaving the "engine" alone, may prove to be better in the long run, rather than authors having to re-write programs every time that the QL hardware is improved.

Don't vanish!

Another problem is that some authors leave the QL scene without depositing their source code with the publishers. This makes it extremely difficult to keep programs compatible with new hardware and results in fewer sales. If you do finish with the QL, please assign the copyright to a reputable publisher and give them a copy of the source code so that they at least have the option of upgrading the program once you have finished with it (or give QUANTA an option on the copyright).

One area where authors can fall foul of both publishers and users is in (lack of) error-trapping. (This non-feature makes itself unpopular even in small printed programs - Ed.) Whether we like it or not, few

of us can guarantee to hit the correct keys every time. Your programs should therefore be protected from any inputs that may disturb or even prevent the running of the program (or indeed if it is multi-tasking any other program running at the same time). Similarly, if you must use HotKeys (that is, where the keyboard is always being scanned for a particular key or combination of keys to activate some command) then always give the user the option of using an on-screen confirmation prompt in connection with each of them (and clear the input buffer before asking for the user's input in response to the prompt) so that users don't find themselves accidentally quitting the program or erasing their files or whatever.

Debugging

Finally there are bugs. Totally bug-free programs are extremely rare, most have a few that manifest themselves in specific hardware/software combinations (and are they then bugs or the result of incompatibilities in the rest of the set-up?) However, it is also true to say that initially many programs will need to be pushed to their limits and tested on a variety of platforms before being released. Clearing bugs can sometimes be achieved by the publisher simply updating toolkit files, re-writing boot programs and so forth. More serious problems will need re-writes in the body of the program. If authors have kept their source code then publishers will have to simply send a list of "bugs" they think they have found (mind you, we sometimes mis-read the manual or attempt things the program was never intended to do!) The authors will then have to try to recreate the bugs on their systems and deduce what is causing them. This can be an impossible task - recently one of our programs was upgraded, but the upgrade only works on certain roms and not others, and we still don't know why! So we have to sell two versions.

The alternative is for authors to send the publisher the source code and let them

arrange for someone else to de-bug the program. This will inevitably mean splitting the royalties, but if it means a better program then hopefully everyone gains. Whichever option you choose, be prepared to spend time on responding to bug reports. This can be quite a protracted process, as it takes time for programs to be bought and used by purchasers and the more unusual hardware combinations may not be encountered for as long as a year after a program has appeared on the market. Publishers will attempt to protect you from too many queries, answering the common or easy ones themselves, but we do sometimes have to pass queries back to authors to see if they can sort things out.

The manual

At this stage the final appearance of the manual will be decided. With a complicated program requiring long and detailed instructions but with low expected sales, it is likely that this will be provided on disk as the cost of photocopying a 40 page manual is quite high, although a guide to using the program may well be printed to provide point-of-sale material for shows, etc. Smaller manuals will, however, need to be provided.

It is vitally important that authors respond rapidly to draft versions of the manual sent to them by publishers as, by this stage, the program should be finished and review copies ready to be sent out. As publishers do not like to dump a newly printed manual just because an author has decided at the last minute to change a vital part of the program, this is likely to be left until as late as possible. Having got what looks like a working and usable program, the publisher will start to advertise it and send out review copies in the hope that this will stimulate sales. Demonstrations of the program at QUANTA workshops can also make people aware of the program.

It is the publisher's responsibility to make sure

that the version that gets sent out for review is as near the final one as possible, as a review focusing on bugs is not going to inspire many sales. (Mind you, a glowing review complete with colour screen shots has been known to result in no sales at all!) Again, it is our experience that reviews can take several months to appear in the QL press, so don't hassle your publisher if one doesn't appear immediately after you've sent the finalised version in.

Money - Money?

So, having completed your masterpiece you now only have to sit back and wait for the money to come in. Well, not quite - your publisher will be expecting you to keep your program operational on all available QL set-ups and so it may be necessary to upgrade the program at regular intervals. And as for the money - well that all depends on how many are sold! Don't expect to get much money during the summer months if yours is a game, as our experience is that very few leisure titles sell well at this time (if at all!) Your other problem is if the publisher fails to make enough money to pay you after paying for the adverts, media, etc. Here you can either take legal action to force them to pay which will probably put them out of business - or you can remind them at regular intervals and hope that sales pick up enough for you to be paid. When you do get paid your royalties, don't forget to declare them to the tax people. And if the program simply doesn't sell - well you'd better get writing a better program!

So you want to be a SOFTWARE AUTHOR

ZX EMULATORS

**Introduced by
Carlo Delhez,
author of ZX81
and ZX
Spectrum
emulators for
the QL**

In the July 1992 issue of *QL World* was an article by Dilwyn Jones about the International QL meeting in Germany. Among other things, he mentioned the existence of ZX81 and ZX Spectrum emulators for the QL. As the author of these two programs, I am writing this article in order to give QL users more information about the emulators.

As most of you will probably know, Sinclair Research had already marketed three other computers before the QL was introduced. In chronological order, these were the ZX80, ZX81, and ZX Spectrum. Although each of these computers had its own unique specification, they were in fact very similar in use, not least because they all contained a Z80 processor, and were operated with more or less the same dialect of Basic.

QL difference

The QL, with its 68008 processor and advanced Qdos operating system did not add to the line of ZX products. As I noticed that many QL users once owned (or still own) one or more of the Sinclair ZX computers, I thought it would be a good idea to start writing ZX emulators for the QL. I started with the job in 1989, and by now two fully compatible ZX emulators are being distributed by me on a Shareware basis: XTricator, emulating a ZX81, and Spectator, emulating a ZX Spectrum. Perhaps a ZX80 emulator will follow in the future.

It is clear that the 68008 processor of the QL is unable to execute programs written in the Z80 machine language. Therefore, the emulators must be able to translate any program written in Z80 code to a set of 68000 instructions having the same effect (from the user's point of view). Over 700 different Z80 instructions exist, and a large part of the emulator code takes care of the translation. Obviously, this translation needs to be done real-time and as fast as possible in order that the emulators

resemble the original computers as closely as possible.

The Z80 processor can address at most 64K or memory. Therefore, a block of this size is allocated in the QL memory, and all actions performed by the emulated Z80 take effect in this area. More to the point, the rom code of any one of the ZX computers can be written at the proper position in this QL memory block and be executed by the emulator as a more or less arbitrary piece of Z80 software.

I/O Instructions

However, the processor emulation described above is not sufficient for getting a usable ZX emulator. For normal computer use, one also needs a keyboard, a screen and a storage medium. The ZX computers usually communicate with their hardware via In and Out instructions. So, another part of the emulators intercepts the input and output instructions, looks as the data and performs the required action, making use of existing QL hardware only.

Firstly, the QL keyboard will be read by the emulator and information about any keys being pressed down is translated to a format that can be presented to the emulated Z80 ports. Secondly, the emulator looks at the (complicated) screen data of the ZX and translates all the text and graphics (in full colour) to the QL screen. Thirdly, any data heading for tapes (or even microdrives in case of the ZX Spectrum) has to be redirected to normal QL files on disk drives.

I could extend this list of hardware compatibility with much more features that are taken care of, but I think the basic idea is now clear. Still, it must be noted that all the above needs to be done in a flash, and be repeated continuously, so that the user really gets the impression of working with the real ZX computer instead of with an emulated one!

Ram/rom sizes

XTricator emulates a ZX81 with 8K rom and 56K ram. Spectator emulates a ZX Spectrum with 16K rom and 48K ram, plus ZX Interface 1 (containing another 8K of rom, microdrive connection, serial interface and network connector). For running the programs, you need a QL with at least 256K ram (though 512K or more is safer), a 3.5in disk drive, and preferably TK2 and the pointer environment (such as QPac 2).

Because of the fact that the Z80 processor is nothing like the 68000, a fast QL is very

important for sufficiently acceptable operation speed of the emulators. Therefore, I recommend that people use the emulators on a QL with a Gold Card (or an equally fast machine) only. Assuming a Gold Card is used (at 16MHz), XTricator can emulate the ZX81 at a speed of about 130% with respect to the real ZX81 in Slow mode; Spectator can emulate the ZX Spectrum at about one-third of the original ZX Spectrum speed.

These numbers are merely average values, because the user can alter the actual emulation speed by playing with job priorities (XTricator consists of three jobs, Spectator consists of four jobs). Both XTricator and Spectator are designed to be fully multitasking. They can safely be used in combination with Qpac 2 and several copies can run in your QL together with other jobs without any problems. Let me now give some specific features of XTricator and Spectator separately.

Features

XTricator emulates a 64K ZX81; tape commands are redirected to disk drives; a set of new commands has been implemented for communication to the QL hardware directly from ZX81 Basic; ZX81 high resolution offered by many games is emulated; ZX81 printer output is redirected to the QL serial port; the QL keyboard is fully mapped on the ZX81 keyboard; a graphical help screen with a scanned image of ZX81 keyboard can be activated by a simple keypress; additional features include SuperBreak, screen invert and Z80 reset, all activated by single keypresses, etc.

Spectator emulates a 48K ZX Spectrum with ZX Interface 1 (issue 2); full colour and flash attribute emulation in QL mode 8; the tape commands of Spectrum Basic are redirected to floppy disk; Spectrum sound can be heard via the QL beeper or via the network socket; of the ZX Interface 1, the Shadow rom Basic extensions, the microdrive commands and the

serial RS232 interface are emulated properly (the LAN will follow later); the QL keyboard is fully mapped on the Spectrum keyboard; two keyboard modes are available (one for serious applications and one for games); a graphical help screen with the Spectrum keyboard can be activated by a keypress; also keypresses are available for SuperBreak, screen invert and Z80 reset; in fact, there are too many features to mention here!

Both emulators are in continuous development (new versions appearing weekly) and have been tested thoroughly with hundreds of existing ZX81 and Spectrum programs (mostly commercially marketed games and utilities written in machine code). The vast majority of programs worked just fine. Only a few programs fail to run on the emulators, but generally these programs are themselves the source of the problem (for example, because of excessive piracy protection), not the emulators.

Currently, 99 out of 100 existing programs will work on the appropriate emulators.

Registration

The emulators are supplied with extensive manuals as 80-column Ascii text files (over 1200 lines per manual). As said before, the programs are distributed on a Shareware basis. To obtain a try-out version of both programs, people may send six DS/DD 3.5in branded disks to me (two for XTricator, four for Spectator). I will always keep half the disks to cover my post and packing costs (I find this a simple solution to that problem). Of course, registered versions can always be ordered immediately. Registration costs 50 Dutch Guilders per emulator. This amount may be sent as a cheque or in cash. Alternatively, I will accept £17 in sterling, or US\$30, or 50 Deutschmarks in cash per emulator. (Comment: Please remember that it is risky to send currency notes through

the post, and in some countries it is illegal. If you do, wrap them up well and don't pin, staple, etc. or draw attention to the contents of your envelope in other ways).

For this fee, people will get the most recent version of the emulator on a 3.5in DS/DD 3M-brand disk, a laserprinted copy of the manual, one update of the emulator for free, and they will be informed about later releases. In the case of XTricator, they also get access to a huge ZX81 software library; the first two disks with 80 ZX81 programs each are dispatched together with the registered copy. For Spectator, they will get a number of disk conversion utilities with the registered copy, which are able to convert programs on original Beta, Disiple and Opus Disk to QL files that can be loaded directly into the emulator. A conversion utility for Spectrum microcartridges is in development. And last but not least: when people register, they support my efforts so that more programs of this kind

may appear in the future!

Finally, I wish to point out that XTricator and Spectator contain the original rom codes of the ZX81 and the Spectrum, respectively. These codes are obviously an essential part of the emulators, but they also are copyright software. Therefore, the emulators may only be used by people owning a real ZX81 or ZX Spectrum with Interface 1 themselves. As it is an impossible task for me to check this for every separate user, the manuals clearly include statements which call the user's attention to this fact; consequently, it is the user's own responsibility when he breaks the copyright law against his better judgement.

For more details, contact Carlo Delhez at Emmastraat 3, 4651 BV Steenberghe, Netherlands. (Anyone who is unsure about the principle of registration for Shareware should read the item on SQLUG in last month's QL Scene.)

plus4 A Quantum leap in QL Wordprocessing

Users have expressed their verdict in letters published in Quanta!

+1 USER FRIENDLY TO THE EXTREME

Context-sensitive help * File selector boxes * Highly compatible with Quill.

+2 THE MOST POWERFUL QL WP

Reformats as you edit * Remembers paragraph formats between sessions * Integrated spell checker with dictionary browse and replace * Multiple Windows over multiple documents allow you to edit one text while looking at a different one for reference * On-screen page-preview and pagination accounting for columns and variable line-spacing.

+3 UNRIVALLED PRINT QUALITY

Nothing else can compete in text and character formatting * Fully supports proportional spacing, mixed character widths and justifies correctly * Multiple paragraph formats with different margins and line-spacing for each * Right, centre and decimal tabs * Multiple columns plus headers and footers * Desktop publishing with several different page layouts in the same document. All the above features available at the highest text-mode letter quality resolutions of 360 x 360 dots per inch (24-pin printers) or 300 x 300 dpi (deskjet and laser) at fast print speed and with very large texts.

+4 FASTEST QL WORDPROCESSOR

Amazing timings on QL and 8 MHz Atari ST with QL emulator. Gold card would be much faster than both * A 70 page text (24,000 words, 141,000 characters) was used * **Load** (first time from floppy disk--not ram disk or slave blocks): 25s (STQL 17s) * **Save** (to floppy disk): 37s (STQL 32s) * **Search / Replace** 580 instances (auto reformats the affected paragraphs to account for different lengths of search and replace strings--not leaving

you with hundreds of lines with ruined format to sort out manually): 43s (STQL 14s) * **Change right margin** of whole text (*with complete reformat*): 65s (STQL 17s) * **Move 10 pages** from top to bottom (*times include manual marking and positioning of the text and reformat after the move*): 35s (STQL 15s) * **Scroll full screen** over text 100 lines (*full screen line by line--not just the last line, not just the last line over every 10th or 100th line of text*): 19s (STQL 5s) * **Change justification** of whole text (*change is instant, times represent full screen refresh after the change*): less than 2s (STQL 1s) * Many other operations (e.g. goto page, line, top, etc., case sensitive search, insert typefaces, auto reformat of edited paragraph, change individual paragraph margins) are **instant**, with only a quick screen refresh to complete the task.

Fully compatible with all QL ROMS, Gold Card, STQL, QVME STE in high resolution, Thor. Requires disk drive and 256 K memory.

Prices (payable by cheque -- air mail overseas)

text87plus4	£79
2488 drivers for 24-pin and bubblejet printers	£19
typeset90-deskjet drivers for all HP Deskjets	£19
typeset90-Epson drivers for Epson Lasers	£39
fountext88 + founded89 graphic driver	£39

Also: QDesign II: £54 * QSpread: £54 * QPac II: £39 * QTop: £34 * QD4: £44 * ArcEd: £24 * FiFi: £19 * Mouse £49

Software87, 33 Savernake Road, London NW3 2JU

DEMO!

We have produced a working demo version of plus4 to try for yourself. The demo has all the commands apart from save, export, spell check and print (sample print output for different printers is included). Send £2 in postage stamps (no cheques please) to get your disk and leaflets.

**Simon Goodwin is a
QL system collector
with an eye on
practical
applications.**

One man's system

My Qdos systems comprise two QLs, one Amiga, one Thor XVI, and a range of other computers including Spectrums and the SAM Coupe. I use these to write articles and develop software tools, so the system is festooned with network, serial and parallel connections to make it easy to share resources.

Samsung QL #1

My main QL is a Samsung model, serial number S-13 003505, made in Korea for the US market, with many production improvements over Thorn EMI's original. The most noticeable difference is a metal coating inside the box, which reduces electromagnetic interference both ingoing and

outgoing, but the board boasts other improvements like an input power filter and modified 8301 display chip, capable of generating 525-line NTSC (US television format) pictures as well as 625-line PAL.

My QL has conventional nine-pin D-type serial and joystick sockets, rather than telephone-type sockets, and a switch on the modulator to select between vhf TV channels. I am grateful to Dennis Briggs of Adman Services for an adapter which lets me connect QL peripherals to the D-type serial sockets.

I also have one of Dennis's replacement QL power supplies, first made by Grundy for the Newbrain computer. This performs as advertised, keeping heat outside the computer rather than next to the microdrives, but I still find that mains interruptions crash my QL more often than other computers, even with a TF Services mains filter and a PC Might 'Uninterruptible Power Supply' en route from the wall socket.

Attempts to take this up with the Electricity Board have been fruitless. When pressed, they installed a chart meter in a wooden box which measured swings in voltage within the ten percent range allowed, and just went clunk, without moving its pen at all, when big supply glitches disturbed the lights and speakers in my flat. The chain

of suppressors usually saves me, but I still need to reset the machine most weeks, and especially on Friday afternoons, for some unknown reason.

Minnie user

The main operating system is Minerva 1.93, although I sometimes swap to older versions when testing software. Like all my computers, this QL lacks retaining screws so I can lift off the bonnet at a moment's notice!

Another recent acquisition is the Hermes IPC upgrade from TF Services. This replaces Sinclair's 8049, improving BEEP, keyboard handling and serial input. It is compatible with all my software, and may encourage me to replace my Sinclair keyboard with the first Schoen replacement, mothballed years ago because of key-bounce problems.

The CST RAM+ memory expansion includes four 32K eprom sockets as well as 512K of fast ram. These sockets suit 16K eproms designed for the cartridge port, as well as specially programmed 32K chips, and give more reliable connection than the QL rom port. At present I have my own *Speedscreen* program, and *SuperToolkit 2* fitted, with room for a DIY Toolkit rom if ever I get around to compiling it.

Standard Speedscreen

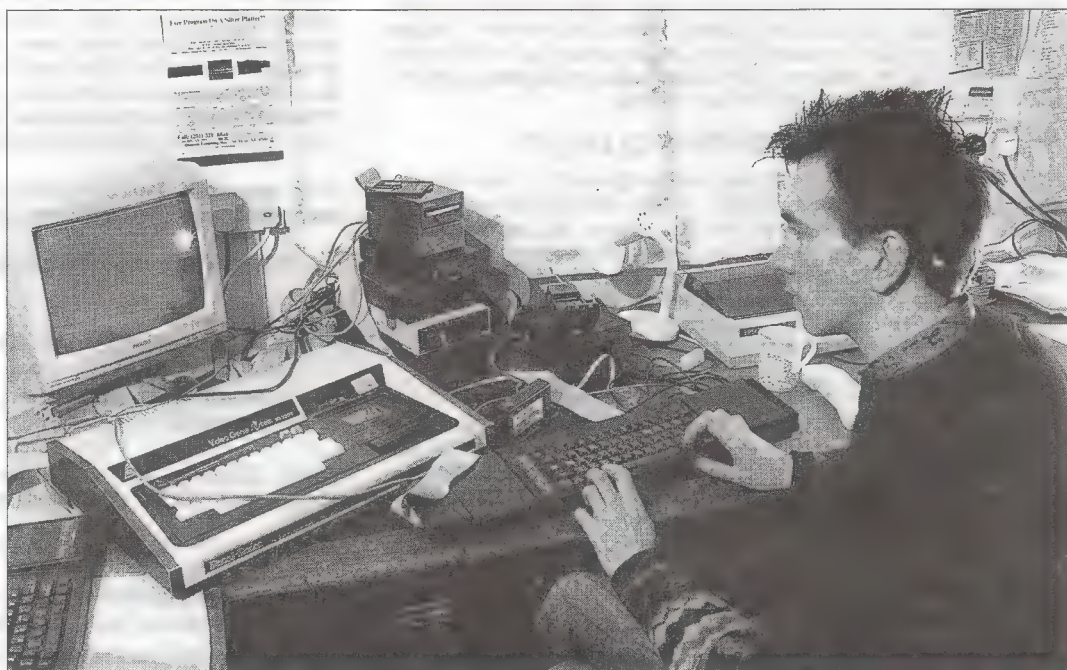
expects the system variables in the usual place at 163840, but one socket contains a variant specially-made for Quanta's P H Tanner, who runs a four-screen network based on two Minerva QLs. This rom is normally invisible, and only signs on if the machine is booted in Minerva two-screen mode.

Big Boot

My BOOT program must suit all my roms, so I use a mix of techniques to detect the configuration and adjust accordingly. **Listing one** is my BOOT program, run from FLP1 whenever I turn on the machine.

The first part loads extension commands, and consists of un-numbered lines. Line-numbers are not desirable, as the program loads toolkit files which are used later; AH and JM roms do not let you use extensions in the program that loads them. The command sequences are un-numbered so they run as they are loaded. They end with a NEW command, to tidy SuperBasic's Name Table and ensure that the extensions are available to the next line.

AH and JM roms ignore eproms in the RAM+ expansion sockets, so the program checks the version and loads Toolkit 2 into ram if necessary. The temporary variable K\$ is needed as direct checks on VER\$ can



crash the "JS" rom, because of a bug in VER\$.

TK2 CODE

The file TK2_CODE is an image of the rom cartridge, made with SBYTES TK2_CODE, 49152, 16384. The CALL sequence locates the initialisation code offset near the start. The network extensions do not work unless Toolkit 2 is in rom or fast uncontended ram.

The program line has been wrapped to save space; the CALL should appear on the same line as the LBYTES command. The next line loads TOOL_CODE, a combined file generated with CUSTOMKIT_BAS, from QL World July 1988. This includes the Qmon 2 debugger, Turbo Toolkit, and many DIY Toolkit extensions.

With these tools loaded I can use extensions to improve the performance of the system. The next line turns on the cursor in channel #0 using Turbo Toolkit. This means that I can type commands as the BOOT program runs, and they will be accepted by channel #0. If this is not done the commands end up in window #1 or #2, where they are not accepted.

The CALL on the same line expands the SuperBasic buffer, as explained here in February 1992. This boosts the speed of COPY and MORE commands. The next line checks to see if the Speedscreen extension "_FOUNT" is defined. If so, it selects top performance rom code with _SPEED 3; otherwise it loads the disk version into ram. The default device is set to FLP1_, and the default for SPL is a SCR window, so COPY FILE uses the display if no destination is specified. I rarely use this since I wrote MORE, which allows scrolling, paging, and fast movement around a file.

The next few lines load more DIY Toolkit extensions, and the TASCOPY_MON command that sends graphics to my old Epson printer. The program checks the Name Table size and reports the number of extensions and free memory in the command window.

```
100 REMark HP Deskjet QL setup from DIP switch defaults
110 REMark PROC TRANSLATE maps QL characters to PC-850 codes
120 REMark v0.8 © Simon N Goodwin 24,25,28-9-92,29-10-92
130 :
140 esc$=CHR$(27)
150 TRANSLATE
160 OPEN #3,"ser" :REMark DeskJet 500 channel
170 PRINT #3;esc$ & "&k2G"; :REMark LF -> CR/LF
180 PRINT #3;esc$ & "&s0C"; :REMark Wrap at right margin
190 PRINT #3;esc$ & "&l26a00"; :REMark A4 paper, vertically
200 PRINT #3;esc$;"(12U"; :REMark PC850 characters
210 CLOSE #3
220 STOP
230 :
240 DEFine PROCedure TRANSLATE
250 LOCAL t$,t,x,ql,hp,c
260 t$=VER$ : IF t$="AH" OR t$="JM"
270 PRINT #0;"This QL ROM does not support serial TRA!"
280 STOP
290 END IF
300 REMark Set simple 1:1 translations for QL -> PC850
310 REMark Beware: NULLS are translated to small blocks
320 RESTORE 490
330 x=RESPR(6+256)
340 POKE_W x,19195
350 POKE_W x+2,6:REMark Table 1 offset on X
360 POKE_W x+4,0:REMark No Table 2
370 t=x+6:REMark Start of 1:1 translate table
380 READ ql,hp
390 FOR c=0 TO 255
400 IF c=ql : POKE t+c,hp : READ ql,hp : ELSE POKE t+c,c
410 END FOR c
420 TRA x,0
430 END DEFine TRANSLATE
440 :
450 REMark DATA list format is [ QL code, HP code, ] 0,0
460 REMark Translations for <NULL>, f, ©, é (coupé), ô (rôle)
470 REMark Slashed "0" needs 48 -> 157, but upsets control codes,
480 REMark so that requires translation in Psion's PRINTER_DAT.
490 DATA 0,254, 96,156, 127,184, 131,130, 152,147
500 REMark Translation for ß (ß test), È (RÉSUMÉ)
510 DATA 156,225, 163,144
520 REMark Translations for µ (µcode), «, », „, →, ↑, ↓
530 DATA 176,230, 184,174, 185,175, 188,174, 189,26, 190,24, 191,25
540 REMark Terminator
550 DATA 0,0
```

Foreign roms

Sometimes I use foreign Qdos roms, with error messages even more obscure than Sinclair's originals, so the remainder of the program replaces the built-in messages with a modified set, using the TRA command introduced in the JS rom. ALLOCATION and INTEGER\$ are Turbo Toolkit functions. One reserves memory, the other converts an integer parameter into a two-byte string. The messages are anthropomorphic, but I like them. You can substitute any text you prefer.

Lines 526 to 532 load and initialise the DIY Toolkit Qlipboard task, which runs in the background and lets me edit small files and transfer text between tasks without re-typing. The task needs to know its own keyboard queue address, so it opens a window and waits for a key-press when first run.

It would be annoying to have to press Enter part-way through, and swap to Basic later, so the boot program saves the SuperBasic key

queue address and turns off the cursor in #0 before loading the task. Turbo's TYPE_IN supplies the task with two enter characters, and suspends SuperBasic for a fifth of a second to give the Qlipboard time to find its queue and select the default window.

Line 532 restores the previous key queue address and re-enables SuperBasic's cursor. This technique can be used to initialise other tasks: for instance, to specify a dataspace for Devpac or Metacomco's ED.

This QL normally serves as a file server for my Thor XVI and second QL. Sometimes I reset the machine because the file server has crashed or locked up, so another machine cannot get through on the network. At line 534 the BOOT program checks for network activity by reading bit 0 of the PC.IPCRD port, which changes rapidly when the network is waiting for acknowledgement. Thus the file server starts automatically if it is needed.

Computermate drive

Since my days writing *Supercharge* I have used a 5.25-in 720K 40/80 track switchable floppy drive, made by Computermate for CST. This reads my old single-density 35 track Genie disks. I have had only two problems - the cable was fitted upside down, requiring a notch in the QDISC socket or removal of the locating lug (I did both) and sometimes I found that disks directories were smashed, rendering track 0 unreadable.

Eventually I traced this fault to the QFLP software, which corrupted the directory when I saved a null file, and then saved another file to the same disk. This was fixed by an upgrade to QFLP 1.16, the last known eprom for this interface, and I have had no trouble since.

I am sure I'm not the only user who finds that their disk drive sometimes keeps running or starts unexpectedly. The disk cable acts as an efficient aerial and

picks up interference. The drive interprets a very short pulse as a signal to turn on the drive. To save disk wear, issue STAT or any disk command; Qdos turns the drive off after use.

Some games cause a similar effect by disabling QL interrupts immediately after loading, so the system does not get a cue to turn off the drive. To cure this, add a short delay between the LBYTES that loads the code and the CALL that starts it - PAUSE 150 should be plenty.

MGT Lifetime drive

My second disk drive is a 3.5-inch, 1 megabyte unformatted 'lifetime drive' assembled by defunct SAM-makers Miles Gordon Technology. It's a good idea: a Citizen disk mechanism with external dip switches and a socket on the outside for cables from a range of computers. The theory is that one drive can serve all your micros.

The drive works perfectly with the SAM, via the SDI interface, and the Spectrum Plus D, answering to the name D1 and formatting disks to 800K. The same settings suit the Amiga, but in this case it appears as DF1, the second drive after DF0, and formats to 880K. Sadly it does not recognise disk changes automatically, so I must issue the Amigados DISKCHANGE command, or Amiga Qdos DSKCNG 1, when changing disks.

The drive reads and writes 720K QL disks, and used to format them too, but recently it has developed a fault and keeps winding back, formatting the same tracks over and over again. This happens whether I use the QL FORMAT command or the Atari and MS-DOS FORMAT options in the ATR device. I use this drive to convert documents into 720K PC format for publication, but have to format the disks on the Amiga Qdos emulator!

The drive also works on Spectrum +3 as a 706K drive B, using the same format as the latest Amstrad PCW model. Dennis tells me that they will give 1.44 megabytes

of formatted capacity on a Gold Card if a link is added on the drive circuit board.

I swap drives often as I work, un-plugging the cables at the interfaces without turning machines off. This has never caused me trouble, but I would hesitate to recommend it in general, because of possible differences in drive connections or stepping speeds, in which case it could crash the system, or worse.

Thorn EMI QL 2

My second QL is numbered D06-006656, and dates from June 1984; the D code is a month number. It was my second QL, as the first only lasted 45 minutes. This is a 128K machine with a QEP-3 eprom programmer in the expansion port, and Toolkit 2 plugged in at the back, so it can read files from the drives on QL 1, and communicate

indirectly with my Thor. A Sinclair printer cable links SER1 to the Amiga's serial port, allowing data transfer in both directions.

Thor XVI

The Thor XVI was my main machine, but it has been sidelined because of power-supply problems. I found that disk access made the display shake, because of a slump in the output of the internal switched-mode power supply. For a while I could cure this by adjusting the preset on the supply, but eventually the voltage fell to the point when the machine would crash soon after start-up. I have a replacement PSU from Dennis Briggs, but have not yet tried it.

The Thor XVI has 6.41 roms, two 720K 3.5-in floppies, and a megabyte of ram. It is linked to the QLs by network, but

eprom files have to be transferred via RAM1 on QL 1, as it will not file-serve reliably for the QLs.

This Thor is currently awaiting a heart transplant; I plan to fit a 20-megabyte Rodime hard disk, the new power supply, and extra circuitry like a disk-control PAL and logic to prevent the SCSI port corrupting the real-time clock setting. Send an SAE to me c/o QL World if you need details of these changes.

Amiga 500

My latest computer is a second-hand Amiga 500, with one megabyte of ram. This runs Qdos almost as fast as the Thor XVI, and more besides. It came with a replacement mouse and drive, and system disks infected by the Lamer Exterminator virus, which caused me some hassle. QL systems are more civilised!

```
REMark Combined BOOT for all QL ROM versions, SNG
WINDOW #0,448,20,40,216:MODE 4:CSIZE 0,0:PAPER 0
k$=VER$:if k$="JM" or k$="AH":a=respr(16384):lbytes flp1_tk2_code,a:
  call a+peek_w(a+6):NEW
a=RESPR(26*1024):LBYTES flp1_tool_code,a:CALL a:NEW
CURSOR_ON #0:CALL PEEK_W(282)+28,1000:REM Expand buffer
IF BASIC_INDEX$("_FOUNT")<0:a=RESPR(11268):LBYTES
  flp1_SPEEDSCREEN_code,a:call a:NEW:ELSE SPEED 3
DATA USE "flp1_":PROG USE "flp1_":DEFAULT_DEVICE "flp1_"
SPL USE "scr_496x210a16x0":ALTKEY 'd','SPEEDSCREEN'
LRESPR TOOL2_CODE:REM BPOKES, QBASE, CLIP, ALTER, TRAPS, PARAMS
LRESPR MORE_CODE:LRESPR DEFS_CODE:LRESPR ALIAS_CODE:LRESPR VOCAB_CODE
LRESPR WINDOWS_CODE
a=respr(2048):lbytes flp1_TASCOPY_MON,a:CALL a
PRINT #0:(basic_l(28)-basic_l(24))/8;" commands, ROM ";ver$;" ";
PRINT #0:FREE_MEMORY;" bytes free, before the 50K RAM disk (RAM1)"
400 k$=ver$ : if k$<"JM" and k$<"AH"
410 x=ALLOCATION(2000)
420 POKE_W x,19195
430 t=x+60
440 FOR a=2 TO 58 STEP 2
450 READ k$
460 IF a<56
470 IF a<44:k$=k$&CHR$(10)
480 IF a=48:READ t:k$=k$&t&CHR$(10)
490 k$=INTEGER$(LEN(k$)) & k$
500 END IF
510 POKE_W x+a,t-x
520 POKE$ t,k$
522 t=t+LEN(k$)+(LEN(k$) MOD 2)
523 END FOR a
524 TRA 0,x:END IF
526 myq=PEEK_L(SYSBASE+76)
528 CURSOR_OFF #0 :EXEC QCLIPBOARD_TASK
530 TYPE_IN CHR$(10) & CHR$(10):SUSPEND_TASK 10
532 POKE_L SYSBASE+76,myq:CURSOR_ON #0
534 FOR i=1 to 200 : IF PEEK(98336) && 1 : FSERVE : EXIT i
536 WTV 4:FORMAT #0,'RAM1_100' : NEW
538 DATA " ***** I hadn't finished!","***** I'm not doing that task."
540 DATA "***** My mind's gone blank!","***** Don't be ridiculous!"
550 DATA "***** I've forgotten what you were saying..."
560 DATA "***** I'm not using that channel.","***** I can't find that."
570 DATA "***** It's already there!","***** I'm already using that!"
580 DATA "***** There's no more data."
590 DATA "***** Not enough room."
600 DATA "***** That's not a command or device name."
605 DATA "***** RS-232 parity error."
610 DATA "***** Sorry - I can't format that.,""Uh?"
620 DATA "***** My drive hurts!"
630 DATA "***** I can't work that out."
640 DATA "***** Cosmic number generated."
650 DATA "***** I don't know how to do that."
660 DATA "***** It's just not listening!"
670 DATA "***** Shome mishtake, shurely?","***** Error at "," lumps"
680 DATA "F1...Monitor","F2...TV","© 1983,1985 SRL (retired)"
690 DATA "I can't trap that.,""I've forgotten what I was doing."
700 DATA "SunMonTueWedThuFriSat","JanFebMarAprMayJunJulAugSepOctNovDec"
RUN
```


Software

The software I use revolves around a collection of SuperBasic programs and Turbo compiled tasks, designed to minimise wasted effort. I write assembler programs, and text files, in Hisoft's *Devpac*. One utility converts code files into HEX data for DIY Toolkit, and another checks the result against the original.

When writing articles I use *TurboQuill* or *Xchange* from the Thor, with a patch to suit Qdos, using the Qlipboard to move things back and forth between Devpac, Quill and SuperBasic.

TurboQuill is fast, but used to keep grabbing the cursor when I used Taskforce to multi-task it; this is no longer a problem since David Moseley told me that the F5 (glossary) option prevents this behaviour. *Xchange* is particularly useful when I'm dividing up files to make sure they will fit on a 128K system; it can EXTRACT blocks to files in _DOC format, and EXPORT plain text.

Parallel Printer switcher

One way or another, my computers can be routed to a parallel printer switch box, which lets me select which computer has access to the printer. This cost £15 at an All Formats Show, plus £5 for the cable to the printer, and saves lots of fiddling at the back of the printer.

My box has three inputs. One comes from QL 1, via Miracle's serial-to-parallel converter, and also serves the QL and Thor network stations. Another suits either Spectrum Plus D or SAM - the cable is the same in either case. The last link comes from the Amiga's Centronics PAR port.

DeskJet 500

The switch box is connected to a Hewlett Packard DeskJet 500 printer, which recently replaced my ten-year-old - but still perfectly reliable - Epson MX-80. Years ago I replaced the Epson rom with a 'Dots Perfect' upgrade, giving NLQ and high resolution graphics rather than the original TRS-80 blocks, but the MX-80 is

obsolete and will not support the latest Epson codes. It was a shame to retire such a faithful servant, but I needed more speed and better graphics without the expense, noise and pollution of a laser printer.

The DeskJet 500 has been favourably reviewed in QL World, so I shall not discuss its merits in detail. Suffice it to say that it offers laser-quality 300 DPI print, and the inbuilt founts match daisywheel clarity as long as you resist the temptation to touch them in the first few seconds after printing, before the ink has time to dry.

Special characters

I shall explain how I persuade the DeskJet to print special QL characters. The techniques involved are applicable to many other printers. Before I use the DeskJet I run the SuperBasic program in **Listing two** to configure it and teach Qdos about its character codes.

Procedure TRANSLATE uses the TRA feature introduced in Sinclair's JS rom to map QL characters onto one of the printer's built-in character sets. By default the DeskJet uses the PC character set, although a few codes are interpreted as controls rather than symbols. Line 200 selects a variation, the PC-850 character set, which has the advantage that it includes the copyright sign, CHR\$(127) on the QL.

TRANSLATE sets up a small table in QL memory, consisting of a six byte header and 256 bytes, corresponding to character codes CHR\$(0) to CHR\$(255). When characters are sent to the serial port they are looked up in this table, and the byte value from the table is used instead of the initial code.

In many cases the QL and HP codes correspond, so the entry in the table matches its position. The QL uses CHR\$(96) for the Sterling (£) sign, whereas the printer expects 156 for the same symbol, so the value 96 bytes into the table is 156. The translation takes place automatically, whatever program is using the serial port.

Other translations involve codes above 127, much less standardised than the seven bit Ascii sequence. The DATA in

Listing two translates the QL codes that I use into PC 850 equivalents, so that I can print E acute, O circumflex, the micro sign (mu) and arrow symbols.

If you use other characters you can add further data pairs. Each pair consists of a QL code, followed by the equivalent for your printer. The list should come in ascending order of QL codes, with two zeros at the end.

This translation scheme is simple and effective but it's not perfect. The DeskJet code for left arrow is CHR\$(27), but I cannot translate CHR\$(188) into that because it is interpreted as ESC! Left triangle, CHR\$(17), is also ruled out as the printer treats that as the flow-control signal XOFF, though the manual says XON. I ended up using the Spanish open quote sign, which at least points in the right direction.

Zero trouble

It is not easy to replace the zero character, CHR\$(48), with a slashed zero, even though that appears as CHR\$(157) in the HP set. The problem is that the zero character is used in many HP escape sequences, when selecting founts and print styles like bold and italics, and these are not correctly interpreted if the zero digits are translated.

I can cure this, when printing from Quill, by setting up a translation in PRINTER_DAT, but this does not affect printout from SuperBasic or other tasks. The translations in PRINTER_DAT are only used when Quill writes text, so they do not affect the codes configured to select print styles.

Both QL and Thor have facilities to define longer translation strings: up to three characters for each QL serial code, or up to 255 using the Thor XVI 'extended translate' facility on SER or PAR. These are only used on output, whereas simple byte replacements operate in both directions.

Philips CM-8833

This medium-resolution colour monitor came with my Thor XVI, replacing the black

and white Pye TV that served as my computer display for most of the 1980s. The CM-8833 is well-specified - it has stereo sound and accepts TTL RGBI (for QL and Thor XVI 8/16 colours), linear RGB Scart (for SAM and Amiga) and even composite video for my old Video Genie computer.

The Genie is now retired, with a broken disk controller, but I can still read all my old files with the Computermate drive and the MULTIDOS software, available from Quanta (Comms Xfer disk 1) and CGH Services (Connections disk 1).

I have no monitor switch box, so I have to tip the display to change from QL to Thor using the DIN input, or from SAM to Amiga on Scart. I leave the Amiga connected most of the time, as the TTL input overrides the Scart socket, so I can swap from QL to Amiga by unplugging the monitor lead at the QL end.

Sometimes a short-circuit changes the QL background colour from black to blue, giving magenta and cyan instead of the usual Mode4 colours. This is because the blue signal, normally unused in Mode4, is picking up current from one of the other pins in the cramped confines of the DIN connector. This quirk does not seem to cause any problems, and Chas Dillon had a similar fault on a TV/monitor for years. If the blue background becomes distracting I can return to the normal palette by tweaking the lead or rocking the monitor, but I should really fix the plug.

Wish list

A QL system is an ever-expanding custom environment, and I have plenty of plans for the future. I hope to have the Thor XVI hard disk working soon, and Dennis Briggs has promised to get me a black serial mouse for experiments. I'd like an adapter to program 32-pin eeproms in QEP-3; the current socket limits me to 64K chips. Most of all, I want a Scart/TTL monitor switcher, but it looks like I'll have to build it myself - my last attempt ended up gummed solid with glue before I got the lid shut!

ARCHIVE YOUR QL COLLECTION

Now you can keep your Sinclair QL World magazines safe and clean. No more dog-eared covers or missing copies . . . You can protect your magazines in this high quality, specially-created binder. This Sinclair QL World binder will comfortably hold a complete year's issues of your favourite Sinclair magazine. It is a high

quality product, British-made and comes with full binding instructions. It is manufactured in a rich, deep blue with genuine gold blocked lettering. Enhance your Sinclair QL World Magazine collection now for only £6.95 (inc.P&P!). Send for one today! The QL World binders also make an ideal gift for other Sinclair users too!

TO: QL WORLD, THE BLUE BARN, WOOTTON, WOODSTOCK, OXON. OX7 1HA

Please send me ☐ QL World binders - I enclose £6.95 for each binder including VAT: postage & packing.

Readers outside the UK and Eire please add £1.95 for surface overseas mail.

Please make cheques payable to **Arcwind Ltd**

ACCESS ☐ VISA ☐ account:

Expiry date:

Name

Address

Postcode

Tel No.

CLUB

FRANCE

QL Contact France. Contact: Jean-Louis Dianoux, 38-40 Rue Stephenson, 75018 Paris, France.

BELGIUM

Club Sinclair BruQsL (Belgium) Contact: Jaques Tasset, Aarlenstraat 104, 1040 Brussels, Belgium

SWEDEN

International QL Conference bulletin board system (Swedish and English). Contact: Michael Cronsten, System Operator, Jamten-TCL, S Soere 1073, 83030 Lit, Sweden.

USA

New England Sinclair QL User Group (USA) Membership Secretary: Sherm Waterman, PO Box 8763, Boston, MA 02114 8763, USA. Magazine: NESQLUG News. Editor: Peter Hale, 195 Central Ave., Chelsea, MA 02150, USA.

NORWAY

Norwegian All Sinclair Association (NASA) Contact: P Monstad, NASA, N-5580 Oelen, Norway. Magazine: Sinclair Magazine.

ITALY

Qitaly Club Chairman: Roberto Orlandi, Via Brescia 26, 25039 Travagliato (BS), Italy. Tel. (local) +39 30 6863311. Magazine: Qitaly Magazine. Editor: Dr Eros Forenzi, Via Valeriana 44, 23010 Berbenno (SO), Italy. Tel. (local) +39 342 492323.

TURKEY

QL Qlub (Turkey). Contact: Bulent Artuz, Prof. Sitesi B/1 D/5, Etiler 80600, Istanbul, Turkey.

AUSTRIA

DER Computer Club. Contact: Peter Postl, Stiegern 5, 1150 Wien, Austria

ENGLAND

Quanta (UK) Membership Secretary, Bill Newell, 213 Manor Road, Benfleet, Essex SS7 4JD. Magazine: Quanta. Editor: Bill Fuggle, 20 Widnes Avenue, Selly Oak, Birmingham B29 6QE.

Bristol sub-group: Roy Brereton, 94 Teignmouth Rd, Clevedon, Avon BS21 6DR.

Essex sub-group: Dave Walker, 22 Kempton's Mead, Potters Bar, Herts EM6 3HZ.

London sub-group: Jeremy Davis, 6 Elmcroft Crescent, Harrow, Middlesex HA2 6HN.

Northern Ireland: Billy Turkington, Fairyhill, Rostrevor, Newry, Co., Down BT34 3BB.

Mid-Southern sub-group: Geoff Fish, 44 Billing Avenue, Wokingham, Berks RG11 4JE.

QL User Group (West Midlands): Mike Bedford-White, 16 Westfield Road, Acocks Green, Birmingham B27 7TL.

QL MUG: 3 Barnard Road, Birkenhead, L43 1TT.

SCOTLAND

Scottish QL Users Group Contact: Alan Pemberton, 65 Lingerwood Rd., Newtongrange, Midlothian EH22 4QQ. Newsletter.

HOLLAND

Sin QL Air (Netherlands) Membership Secretary: Bob Visser, Snelrewaard 6, 2904 SN Capelle, a/d IJssel, Netherlands. Magazine: Quasar. Editor: C H M Biemans, Elzenstraat 5, 5461 CL Veghel, Netherlands.

GERMANY

Sinclair QL User Club eV (Germany) Foreign Contact: Franz Herrmann, Talstrasse 21, d-W5460 Ochenfels, West Germany. Magazine: Quasar.

SPAIN

Qliper Editor: Marcos Cruz, Acacias 44, (Montecarlo), E-28023 Pozuelo de Alarcon, Spain. Magazine: Qliper.

In the last of his occasional series on desktop publishing, Geoff Wicks tackles the serious business of editing fonts.

One of the paradoxes of desktop publishing is that although you use only a minority of the fonts supplied with the program, you are always wanting more. The reason for this is that the suppliers of DTP programs have to cater for many different users. Some users want to produce eye-catching advertisements and need fancy display fonts; others are preparing overhead sheets for illustrating lectures and need sober display fonts; and yet others are producing newsletters and reports and need text fonts and display fonts suitable for headlines. Some users will be satisfied with a simple printout of their document, others will reduce their document at the photocopying stage and yet more will use reduced width and length printing. All these users require a different range of fonts.

In this situation it is practically impossible for a software publisher to cater for all needs. Designing fonts is very time consuming, and the

fonts require large amounts of memory and storage space. To keep the costs of a program within the reach of the potential users a software publisher must compromise and this means that the fonts that are supplied are often not quite what you would ideally like. For this reason one of the essentials of a desktop publishing package is a good font editor so that you can design your own fonts or adapt the supplied ones for your own use.

Editing fonts

I suspect that there are many users of DTP programs who never look at the font editor and are perhaps even frightened of the thought of attempting to modify a font. This is a pity because once you get the hang of things editing fonts is quite simple, although there can be a lot of time consuming, humdrum work involved. The secret is to keep things as simple as possible. It is better to modify an existing font than to attempt to design a new one and to start with the smaller fonts rather than with a large one.

If you have never edited a font before I am going to guide you through the stages. Firstly I am going to break my golden rule that there is no place in DTP for low resolution QL fonts. When learning to edit fonts you can best begin by modifying the standard QL

font, because the matrix on which it is formed is small so that the font can be very easily edited. Within a hour or so you can learn the principles and the potential problems of font design which also apply to the high resolution fonts.

What we are going to do is to make a wide version of the standard QL font. First we need a copy of the font which we obtain by running the following short program:

```
10 addr = PEEK_L(PEEK_L(
PEEK_L(163960)+4)+42)
20 SBYTES flp2_QLfont,ad
dr,875
```

This font can now be loaded into the QL font editor of your desktop publishing program. Select A as the character to be edited. In the far right hand column of the matrix fill in pixels in the places where they appear in the right hand edge of the letter; fill in the horizontal pixels to join the old and the new edges; and finally remove the pixels forming the original edge (see **Figure one**). The letter is now eight pixels wide instead of six. Store the letter and then move on to B.

Surprisingly quick

In this way you can work through the whole of the character set, and once you are used to it you will be surprised how quickly it can be done. The letter I has to be

A Question of Fonts

Part 5

altered in a slightly different way, and before you start on K save your work so far. The

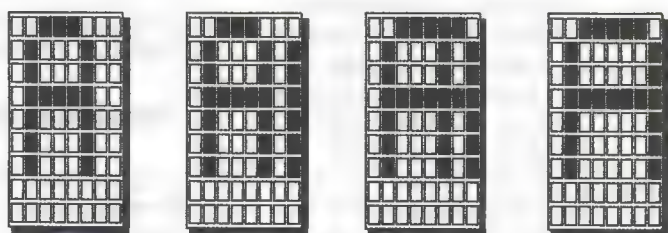


Fig. 1: Modification of the standard QL font.

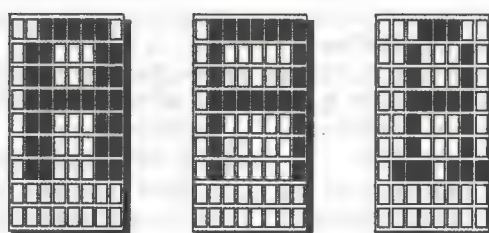


Fig. 2: Bold, Square and Serif letters.

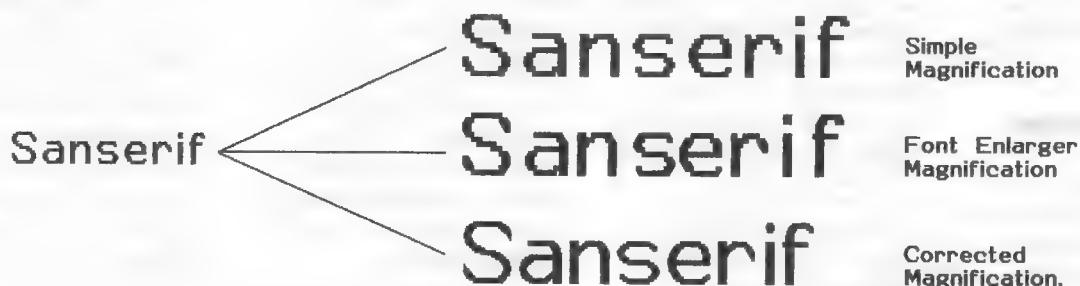


Fig. 3: Magnification of High Resolution Font using the Font Enlarger Programme.

reason for this is that some letters are more difficult than others and K is one of the problem letters, as are most letters containing a diagonal.

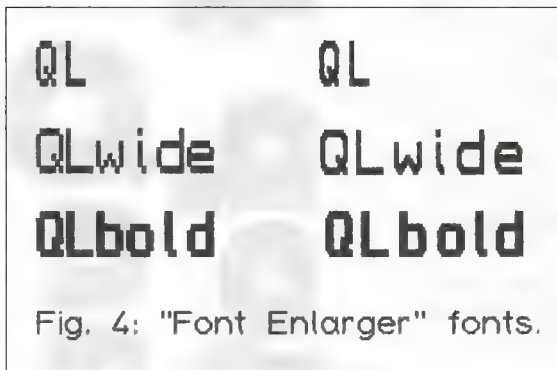


Fig. 4: "Font Enlarger" fonts.

In many fonts, especially high resolution fonts, R can be difficult to design and in almost all cases Q. With these letters you must be prepared to experiment until you get the best result. If you get stuck you can always examine other fonts to see how others have overcome similar difficulties.

You will also soon discover that it is good practice to save a font regularly when working on it, especially in the case of high resolution fonts. There is nothing more frustrating than to find that having successfully designed 15 letters you get hopelessly stuck on the 16th. If you had saved the font before the 16th you could always reload the partially completed font and start again on the 16th without losing your previous work.

When you have completed your wide version of the QL font, save it. Now try modifying it into a bold, square or serif version (**see Figure two**) (Hint: when making a bold font you get the best results by just making the vertical lines of a letter bold and not the horizontal). Again you will find most letters can be modified quite easily and it is just a handful which cause problems. You will soon learn which letters are difficult and of possible ways of modifying them.

High resolution

You are now ready to move on to designing high resolution fonts. Again the advice is to start simply with one of the smaller fonts. If you choose *Professional Publisher's*

Fine_Big font, which has a 48 x 48 matrix as your first font for modification you will soon give up in despair. It is better to try modifying *Excellent* which has a 16 x 16 matrix and which has a good basic letter shape. Try, for example, to make a widened version of it. The widened version is a good font

for text use if you use reduced length and width printing. Another good font is *Grotty*. If you want to be more ambitious you can try modifying *Roman* into a sanserif version. (In earlier versions of *Professional Publisher* the *Excellent* and *Roman* fonts are named *Helvetica* and *Times* respectively.)

When you have gained a little experience you will find yourself looking critically at different fonts in magazines and desktop publishing programs for other computers or even start looking at the fonts used on road signs and in such places as hospitals and railway stations. You may even look at ways of converting fonts from other sources into a suitable form for *Professional Publisher*. For example NLQ fonts which are suitable for downloading into dot matrix printers have a matrix size which makes good fonts for *Professional Publisher*. The fonts do need converting to *Professional Publisher* format.

On the recently published *Professional Publisher Toolbox 2* disk are a number of fonts designed by Sohail Bhatti which perceptive members of *Quanta* will recognise as being similar to fonts available from the *Quanta* library which he has designed for downloading into the *Star NL10* printer. I have used one of these fonts for the captions in the illustrations for this article. The *Toolbox 2* disk also contains a program for converting *QWRITER 2* fonts into *Professional Publisher* format.

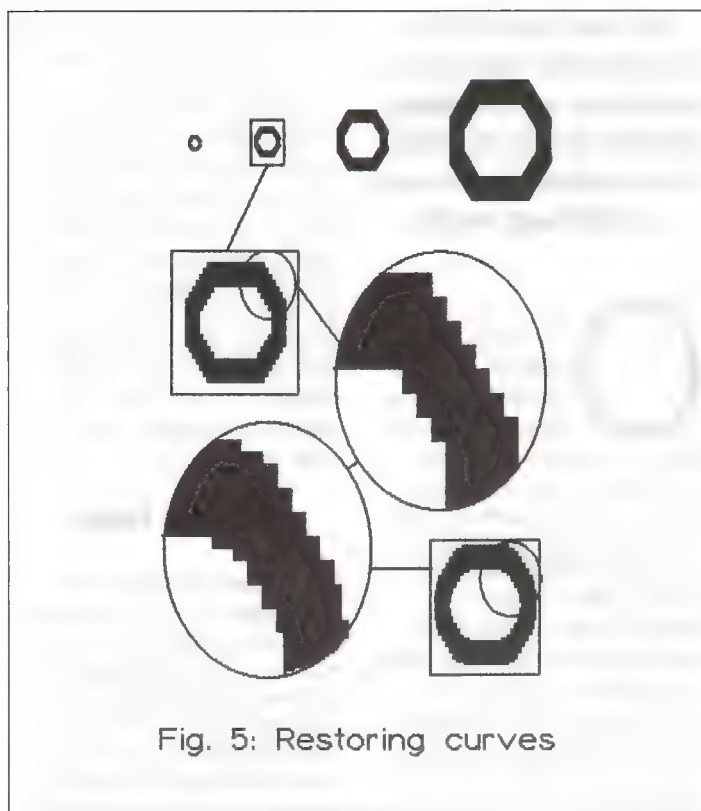


Fig. 5: Restoring curves

Widths and tabs

With high resolution fonts you also have to adjust the width of a letter. An 'i' is much less wide than an 'm' or a 'w'. This is a fairly simple process and is described in your DTP program manual. Proportional spacing is essential for the larger fonts but one disadvantage of proportionally spaced letters is that tabs do not work on text imported from *Quill* or other editors. This can also give problems in documents illustrating arithmetical calculations. For this reason it is sometimes better to give all the numbers in a font the same width.

It is possible to produce a font in a number of different sizes or to make a high resolution version of a QL font by using Mark Knight's *Font Enlarger* program which is available from Digital Precision. This program is very easy to use and has the advantage that you can design a font on a small matrix and then enlarge it which is a lot easier than designing it straight away on a large matrix. This program does have its limitations however and this I have illustrated in **Figure three**.

If *Font Enlarger* did nothing more than simply magnifying a

font, then the enlarged font would have ugly steps. Instead the program attempts to do something very difficult and that is make a high resolution enlarged font, which means that it has to make an intelligent guess about where to add or take away pixels. Most of the time it does this very well, but sometimes it makes a mistake. The *Font Enlarger* produces a better font than by simple magnification and in most cases you could get away with using it without further editing. However I am one of the purists as far as font design is concerned, and in my opinion you should always smarten up the font by a little editing.

Learning curve

In **Figure three** I have deliberately chosen a font and a magnification which tests *Font Enlarger* to its limits and thus emphasises its shortcomings. However to redress the balance in **Figure four** are samples of three high resolution fonts (right) produced from low resolution QL fonts (left). It is quite clear that in these cases practically no further editing is required.

Another problem with font enlargement is that when a small font is magnified, perfect curves are not produced. In

this way a 'O' starts off as circular and ends up as a octagon. This is another reason for editing fonts after using Font Enlarger. I have illustrated this in **Figure five**. The top row shows the effect of simply magnifying an 'o'. The second 'o' is shown magnified and one corner of this is further magnified to show how the curves can be restored. This done by removing pixels on the outside of the letter and adding pixels on the inside of the letter. (Look at the change in the size of the 'steps' in the two circles showing the enlarged portion of the 'o')

One final hint.

Designing fonts is one of the more tiring forms of computer use. In particular it gives more than usual strain on the eyes. For this reason a good monitor is needed and it is better to spread your work over a number of short sessions than to work through a high resolution font in one go.

I hope that this hint together with my comments about humdrum work will not put you off having a go at editing a font. Indeed if your desktop publishing is not in English you will have to do some font editing to provide letters with accents, umlauts, and so on. This is very easily done with Professional Publisher and it is not necessary to modify the whole font. You can just design those letters in the Ascii range 128 -191 which you need for your own language. For your work and your patience you will be well rewarded.

Finally in **Figure six** are some of the text fonts I have designed based on existing fonts. The top four source fonts are low resolution QL ones and the bottom two high resolution fonts.

Let us know

This article brings to an end this series on desktop publishing. In the past I have been disappointed by articles

Source Font	New Font
Standard QL	Standard QL Thin QL Serif QL
Avant Garde	Avant Garde
Olde	Olde
Classy	Classy Classy Thin
Helvetica	Helvetica Wide
Times	Sanserif Sanserif Bold Serif Serif Bold

Fig. 6: New fonts from old fonts.

and examples of QL Desk Top Publishing which have appeared in the QL magazines and have tried in this series to encourage the user to attempt to get more out of their programmes. I hope I have succeeded.

It would be interesting to know what use QL users make of DTP, to see some of the output and to hear what problems they have come against. Letters to *QL World* would be very welcome.

TF Services

MINERVA

The ULTIMATE operating system upgrade

MKII MINERVA now with battery for 256 bytes ram, CRASHPROOF clock & interface (I/O) for robots etc. Can autoboot from battery ram.

Other features (common to MKI/MKII):
Autoboot on reset/ Multiple Basic/ Faster scheduler-graphics (18% of Lightning)-string handling etc/ WHEN ERROR/ Second screen/TRACE Foreign kybd drivers/ "Warm" fast reset etc

MKI to MKII UPGRADES: up to 1.79-£5. Rest FREE (send rom (PROTECT LEGS!) and disk or 2 mdvs)

Original MKI..... £40 MKII.....£65

ALL VERSIONS FULLY GOLD CARD COMPATIBLE

BULLETIN BOARD

QBBS - The first UK QL scrolling bulletin board. Now even larger with international ECHOmail and 10mb of files (including the WHOLE King James' bible!)

071-706 2379 M21/22/22bis 24hrs

Qualsoft

Viewdata/VT52/VT100 QL TERMINAL EMULATOR

Multitasking for electronic mail, PRESTEL etc, CET download. Compatible with Tandata Qconnect QL modem/adaptor. Use Qconnect RS232C to drive unbuffered modems. Comprehensive RS manual.

Software (3.5" or mdv) & manual £30

FILE TRANSFER - to transfer text AND programs error free (XMODEM) between computers

Available for IBM compats, ATARI ST & Sinclair QL Connects to Psion range with RS232C comes link Software £7.50 per m/c. Cable £18 (name 2 m/c)

All prices include VAT and postage in the UK

HERMES

A replacement QL co-processor (IC 8049 IPC)

Do you get keyboard bounce?
Do you find Fast serial input unreliable?
Do you want to run an RST 16000 bps modem?

If you can say YES to any one of these, then this is a product you have been waiting for

- Gives up to 19200 bps RELIABLE serial input
- Independent baud rates
- Stops keyboard 'bounce' (unwanted repeat chrs)
- Improves 'fuzzy' and 'random' sound
- Provides extra input and output lines
- Key click
- Reset/INT7 invoked more safely

FITTING is a simple job. Simply remove the QL top (8 screws) and replace the chip marked '8049' next to microdrive 1.

PRICE (including manual & software) £25

QL SPARES

Faulty QL boards (excl plug in ICs)..... £9
Keyboard membrane... £9 Circuit diag... £2
68000 cpu..... £8 1377 PHL..... £3
JH rom set..... £10 Power supply... £12
9302 ULA..... £10 9301 ULA..... £10
8849..... £8 MDV ULA..... £12

Other components (sockets etc).... please PHONE

QL REPAIRS

QLs tested with Thorn EM1 test rig and ROM software
(MDV hardware extra if required)

£27 - 6 month guarantee

MAIL ORDER ONLY

12 Bouverie Place, London W2 1RB (tel: 071-724 9053)

SCROLLING BBS FOR ORDERING/QL HELP and FAX - 071-706 2379

QUBBESoft P/D

38, Brunwin Road, Rayne, Braintree, Essex. CM7 5BU. Tel. 0376 347852

Public Domain & Shareware Software

C68 Compiler Runtimes (3Disks) £3.00p
C68 Compiler Src Cde (2 Disks)... £2.00p
MicroEMACS V3.11 (2 Disks)..... £2.00p
GNU Text Utilities (2 Disks)..... £2.00p
QL Emu for AMIGA (2 Disks)..... £2.00p
SPEC 1 Q.A.T.S..... £1.00p
SPEC 2 Molecular Graphics..... £1.00p
SPEC 3 XLisp V1.4a..... £1.00p
SPEC 4 QPacer..... £1.00p
SPEC 5 C Tutorial..... £1.00p
SPEC 6 Lib QPTR..... £1.00p
SPEC 7 Lib CPORT..... £1.00p
SPEC 8 Q/Ractal Screens Demo..... £1.00p
SPEC 9 QL Home Finance..... £1.00p
SPEC10 ZM+ SPECTRUM Emu..... £1.00p
SPEC11 Lib CURSES..... £1.00p
SPEC12 ELVIS Editor..... £1.00p

PD Disks 1 to 11.....£1.00p ea.
(Each Disk contains 10 programs and a Menu Handler. Each program can be RUN from the MENU unless other-wise stated).

New QL Hardware

QEP 3 EPROM Programmer:- Will program 2716 to 27512 5V MOS & CMOS Eproms. A4 Manual Inc. £120.00p

Trump Card :- 768K RAM, Toolkit 2, Disk Interface, Dynamic RAM Disk, A4 Manual. £95.00p

ExpanderRAM:- 512K of RAM with thru port to connect other peripherals (i.e. Disk I/Face, Eprom Board). £45.00p

ExpanderRAM:- 0K of RAM, as above except User supplies RAM chips (8=256K, 16=512K). £20.00p

Twin 3.5in Disk Drives :- Fully cased with internal P.S.U., 720K formatted capacity per Drive. £100.00p

2nd User Hardware & Software

QLs (JS/JM).....From £45.00p
Disk Interfaces.....From £30.00p
Disk Drives (Twin).....From £75.00p
Printers.....From £55.00p
Monitors (Grn).....From £35.00p
Monitors (Colour).....From £75.00p
Used MDVs.....80p each
QL Related Books.....Half Price
QL World Back Iss.....75p each
QUANTA Back Iss.....40p each
All types of Commercial S/Ware including Games, Utilities, Front Ends etc. etc., too many to list.

Payment Methods

Cheques (in UK Pounds Drawn on UK Bank), Postal Orders, International Postal Orders payable to:- QUBBESoft P/D

Eurocheques payable to:- R.Dunnett

Cash in UK Pounds is also accepted.

All Prices Exclude P & P. Please Ring for details

P&P PD & Shareware

Qty.	U.K.	Euro	Z1	Z2
1-3	£0.50	£1.00	£1.50	£1.75
4-6	£0.75	£1.50	£2.00	£2.25
7-9	£1.00	£2.00	£2.50	£2.75
10-12	£1.25	£2.50	£3.00	£3.25
13-15	£1.50	£3.00	£3.50	£3.75
16-18	£1.75	£3.50	£4.00	£4.25

Euro-All EEC & Scandinavian Countries
Z1-Middle Eastern, Nth & Sth America & Canada
Z2-Australia, New Zealand, Japan etc.

Catalogue

S.A.E. (A5 Size) or International Reply Coupon.

Sinclair/QL World December 1992

41

Perfect Paragraphs & Shining Characters

Part 3

In the final part of his proportional printing project, Martin Harris analyses rules for deciding which errors you want reported.

Last month we went through examples of all the type code errors you are likely to meet in layout out a document. This month I will consider how one decides which errors the program should correct, and which it should simply report for consideration.

In my particular case, five operations are actually carried out by the program: underline and enlarged corrections, hierarchy adjustments, backspace accented characters, and proportional insertions. Only the first of these may be a genuine error. On the other hand, many more errors are merely noted and advised. To have our program physically correct all these would involve imparting the computer with a degree of intelligence both well beyond the scope of this article, and definitely not worth the effort in terms of time saved.

Decisions, decisions

We must therefore decide what kind of information we want to report, in order to best adjust our document manually. I chose the code type and the line number. You might also want to add the error type, such as missing end-toggle or density change error, or you may be happy with just the line number.

An integer array is an obvious vehicle for remembering these errors. As I have not yet updated my program to include rubber arrays, the array must be dimensioned to a specific size. There are seven code types and I chose 10 errors per type as being on the safe side (lines 1220 and 1260/1270). If however there

were more errors of any one type than this, only the first ten would be reported. Since each error type can be represented in the standardised form of code and line number, it would seem appropriate to write routines to cover them all. In an earlier version I had not understood this point and had separate routines for each error type. The result was a program about 50 lines longer.

The routines "note_error" (lines 3790 to 3830) and "tell_error" (lines 3840 to 4020) are fairly self explanatory. "Note_error" is called whenever an error is detected and fills the error_array with the line number and type of the error. "Tell_error" is called by line 3660 in a short loop and uses the small window created at the start of the program. To avoid an unduly large window, it pauses between error types until you press any key by the procedure "wait" (lines 3590 to 3630). Note also that sometimes the line number fed into the procedure "note_error" is sometimes lno - 1 and sometimes just lno (eg lines 2770 and 2930). This is because where an error is detected at the end of a line, the line number is automatically incremented to the next line before the error is noted, (line 2680) therefore this situation has to be adjusted back.

Proportional toggles

We now come to the tricky job of inserting proportional toggles into a document. First, however, we must analyse why we make certain choices and not others, before we can even begin to tell the computer how to do it. Consider the following examples:

EXAMPLE 1

1. This is an example of random text. This is an example of random text. This is an example of random text.

No doubt you would insert proportional toggles as follows:

P1. PThis is an example of random text. This is an example of random text. This is an example of random text.

EXAMPLE 2

1. Beneath this title text is real text. It's just waffle.

This is the real text referred to. It's just waffle.

Proportional toggles work out as:

- P1. Opposite this Ptitle text is Preal text.

PThis is the real Ptext referred to. PIt's just waffle.

EXAMPLE 3

1. This is another example of random text. This is another example of random text. This is another example of random text. This is an example of random text. This is an example of random text.

No proportional inserts here. But the first line of this example starts in the same way as for example 1.

EXAMPLE 4

1. This is an example of random text. This is an example of random text. This is an example of random text.

1.1. First sub-paragraph one line text

1.2. Second sub-paragraph one line text

In this example, no prizes for guessing where to insert proportional toggles. There is however a difference with the other examples, in that there is a blank line between 1.1. and 1.2. Supposing further that there were two blank lines in between? How many blank lines in between would it take before you decided that no proportional inserts were necessary: 5, 10, the next page? Are you being consistent about such decisions from one document to another?

And what about the following:

1.1. First sub-paragraph
one line text

Some intervening text

1.2. Second sub-paragraph
one line text?

Another question about your
own decision making:
consider the following slightly
altered version of example 2:

1. Opposite this)
title text is)
real text.)

This is the real
text referred to.
It's just waffle.

and compare it with
example 1:

1. This is an example of
random text. This is an
example of random text.

Is there any reason why
example 1. should not be
dealt with as follows:

P1. This is an example of
random text. PThis is an
example of random Ptext.

Obviously, there's quite a lot
of grey in how we make our
decisions. Let us try to simplify
things a little and ask
ourselves the lateral thinking
question: are there any areas
we could imagine where even
we would not be sure about
what to do? If we can say
definitely no or almost never,
then no matter how complex
the rules might be whereby we
made such decisions, such
rules must exist and be clear-
cut. I am always sure where to
put proportional inserts, but
even if I were not, such
circumstances would only
concern the most unlikely or
remote possibilities. In such
cases, there is nothing to be
lost whatever decision is
made.

Suppose, though, that some
of the decision-making was
based on my knowledge of
English (as it might be in the
last example question posed);
it is out of the question to
teach my computer such
rules. In general, therefore, we
cannot hope to have an
identical reproduction of our
decision making and must
settle for an acceptable

difference, with a tendency to
err on the safe side. And so
long as we ourselves are sure
why we do something, we will
improve the computer's
performance to the maximum.

The rule book

These then are the rules,
which, after considerable
thought, represent about 98%
of the basis of my own
decision making in this
particular system:

1. There are aspects within
each line which determine
whether insert(s) should be
made, such as the presence
of a close bracket or a dot or
other similar character which
may indicate that the next
printable character after a
following space should be
preceded by an insert (and
correspondingly at the start of
that same line).

2. If, and only if such
aspects correspond to
aspects in identical positions
in lines above or below the line
in question within a
reasonable distance away, will
such situations qualify for
proportional inserts. I also
found that the number of such
correspondences had to be at
least 2, otherwise too many
spurious inserts occurred.

3. The "reasonable distance
away" varies according to how
far along the line one is. For
example near the start of a
line, an insert may be
triggered by another aspect
up to 40 lines above or below.
This has to be so to allow for
long sub-paragraphs.
However, near the end of a
line only four lines above or
below should be looked at.

4. Proportional inserts
should only be made when
proportional mode is on and
should never be made when in
compressed mode.

Once filled out with detail,
these rules correctly
analysed 98% of inserts.
Some of the errors were
inserts not made, but most
were inserts made which
should not have been.

What size net?

The "reasonable distance
away" and other such ideas
act like a net with different
mesh-sizes. If you pitch the
size too coarse, you lose
things which should have
been trapped. If you pitch it
too fine, you catch things
you didn't want. Most of the
inserts which I caught but
which I didn't want were the
result of looking 40 lines
above and below up to the
15th character on the line. In
these cases it didn't look too
bad if non-proportional
mode was on by mistake for
just a few characters in from
the left margin and then only
two or three times in a long
document.

There is some capital to be
gained from this type of
analysis in that the computer
can help you to change your
own habits. Not that we want
to be dictated to by the
machine but perfectly
reasonable suggestions may
come to light in this way. I
had noticed that often the
computer would make
spurious inserts, even near
the end of the line. This was
triggered by certain
characters and a space;
however, I had to leave this
combination in since very
often at the start of
paragraphs this would truly
give rise to an insert (such
as in the examples quoted
above).

However, I decided that I
would limit myself such that,
except for the initial
paragraph number or letter
itself, I would have to put in
two spaces instead of one, if
I wanted something to line
up in the middle of a line.
This was in any case good
practice.

Rules to codes

Converting these rules to
computer language is quite
fiddly but without real
problems. In our case, this
falls into certain distinct areas:

1. Create an array to store
information concerning rule 1.
I chose five possibilities per
line, including the first
printable character (line 1530).
2. Fill such an array with
information obtained from a

separate line analysis loop
(lines 1750 to 2400). This is
quite a complex operation
logically and space does not
permit me to explain it in
minute detail here. Basically it
picks up certain characters
(including spaces) followed by
spaces and assigns the line
position to the array, with up to
5 possibilities per line.

3. As the main program loop
(already discussed) is turning
round, check to see if rule 2 is
applicable by analysing
forwards and backwards
through the array. This does
not need to be done at every
line. Rather, the function
"analyse" returns the next line
where inserts should be made
and only needs to be called
when the loop has dealt with
the current insert (lines 2600
to 2660). If this is an
applicable case, do the insert
at the start of the line (line
2620), and in the middle of the
line at the position marked by
nltab (line 2630) having regard
to rule 4 (line 2610).

4. The function "analyse"
takes care of rules 2 and 3,
returning the line number of
the next line to have
proportional inserts added
and updating the value of
nltab (line 3520), which
remembers exactly where on
the line the second toggle
should go (the first is
automatically the first printable
character on the line).

Conclusion

The program should of
course be compiled. Don't
attempt to test it in Basic, as it
will tie your machine up for a
very long time! Testing is in
three stages:

(i) Load the program into the
Superbasic editor: if it then
contains MISTakes, correct
any syntax before proceeding.

(ii) Compile it with line
numbers and correct any
errors detected by the
compiler. These are usually
missing END FORs and the
like.

(iii) Run the program many
times under test conditions,
noting where it crashes and
correct these errors. Check
also the results of the test

**Bryan
Davies goes
to Quanta
and tracks a
missing user.**

shooter

Trojan

Software87 has produced a demonstration disk for *Text87 Plus-4*. The price is £2 (in the form of postage stamps). The version number of Plus-4 itself has reached 3.1.

Miracle Systems have made some detail changes to the Gold Card. The rom version has reached 2.31. The latter is said to differ little from 2.28, and has been introduced for version 3 of the pcb; it is compatible with the earlier version 2, and the original yellow pcb. Users fitting replacement roms themselves should take note of the position of the notch in the end of the existing rom before removing it, as there is no apparent guide to which way round to fit the replacement. The notch should be at the end which is adjacent to the Ingot chip - that is, NOT at the edge of the pcb. The values of the two blue capacitors at the edge of the pcb alongside the rom have been swapped; the one nearest the edge is now 15 nF and the one behind it is

33 nF. The change is understood to be related to problems with the handling of HD disks; there doesn't appear to be any need to switch the capacitors around if you have no disk-handling problem.

Poking about

While on the subject of the Gold Card, Miracle commented on the subject of using the POKE to speed up Gold Card operation: "If it doesn't work, you are lucky!" This POKE is a command which can be issued from the SuperBasic command line, to give an apparent big boost to performance. Type:

POKE 114796,0 then press ENTER

The idea is that this apparently simple command boosts the Gold Card clock rate from 16 MHz to 24 MHz, a nominal 50% performance increase. Some people have reported obvious improvements, but the effect of using the command in some QL systems is to lock them up. That happens on the two QLs I have tried it on, so I am in no position to comment on any performance gain. Miracle feel that the unpredictability of the effects produced by this command make it an unreliable thing to use; they prefer that users stick to the already-fast 16 MHz of normal Gold Card operation.

One program which should benefit appreciably from use of the POKE mentioned above is Digital Precision's *Conqueror Gold SE* (which has reached version 3.00). In emulating another computer and operating system wholly in software, every little booster is worth having. If you can't make use of the POKE, maybe you can speed-up program loading by using ED (extra-high density) 3.5-inch drives. One ED disk can be used to hold the one file necessary for the complete "DOS

environment". This is created the same way as it would be with a hard disk drive, and you get what is effectively a 3.2 MB hard disk on one ED floppy.

Conqueror

For users who have bought Conqueror with the DR-DOS 6 operating system, there is a further bonus, in the form of the disk-compression routine SuperStor, which will make the effective size of the single floppy file into 6.4 MB. In practice, you don't normally get the full 2:1 increase in available space, because different types of file permit different levels of compression. Data files usually have more "blank space" than program files and can be compressed nearer to the 2:1 mark, but an average of 1.4 to 1.8 can be expected. Having said that, it must be stated that DP report getting over 8 MB!

The Gold Card's 2 MB of ram makes the running of extra-large programs much more satisfactory. Improvements to Conqueror have yielded, amongst other things, much more available DOS memory. Whereas you used to see a memory allocation figure of 667 KB with the previous version, the 3.00 version boots up with the message "Extended memory allocation (KBytes) is 944". Don't get carried away, though. The figure is not the amount of additional conventional memory, but it can be used by programs which support Extended memory - such as the DOS VDISK (ramdisk). The actual available DOS space shown after boot-up was 670 KB, which is about 40 KB more than I've been able to obtain using all the tricks on a PC. To improve disk access times, you are advised to leave 30-40 KB of the Extended memory available for Qdos, to use for

slave blocks; that is, set the Extended memory figure to about 900 (even 850 if you can spare some more).

In operation, booting of the emulator from ED disk is appreciably faster than it is from DD disk. It takes under 30 seconds on my system, using fairly simple boot files. The initial text ("Loading...") appears on the screen almost immediately, making it clear that one needn't settle down for any thumb-twiddling now. WordPerfect 5.0 ran at a decent speed, from DD disk; subjectively, it felt appreciably brisker than it did when using the earlier version of Conqueror. With application programs on ED disk, and a ram disk set up for overflow files, performance would be still better.

Still active

The heady days when QL activities took up a large part of the hall at computer fairs are long gone, and it is easy to think that most of the group QL activity has died out. The recent first meeting of a newly-formed local Quanta group was a bit of a surprise, as about a dozen and a half people turned up, and the atmosphere was much as it was several years ago. Most people there were fairly old hands with the QL, but a few were new to it, and in need of basic advice. No surprise that help was needed on printing from *Quill*! Equally unsurprising was the difficulty I had in remembering how to use Quill, as it is something I normally don't touch. It still remains to me the easiest of programs to learn (and remember), though.

One thing was apparent from even very brief discussions with fellow members - the talk was not QL-only. Whereas most of us would have had just the QL a few years ago, many have one or more other types of computer now. To a large extent, this is a reflection of increasing affluence over the years, coupled with the big drop in prices of computer equipment in general. It is cheering to see that the attractions of the QL remain

strong even to people who have been using other types of computer for some years. All in all, it was a successful evening, thanks to the enthusiasm of prime mover Joe Hafke (who wrote some of the programs distributed by Dilwyn Jones Computing).

A topic that was brought up (by Tony Firshman of TF Services) at this meeting was that of computer fairs. Although the venues in southern England used for the All Formats Fairs must be within reasonable reach of some QL owners, it seems unlikely that the bulk of those who used to go to the central London venue will feel it worthwhile to trek to places such as Sandown Park (in Surrey). The prospect now is of no common meeting place for the general run of QL users, traders, and Quanta representatives. Some local Quanta group meetings will be large enough to attract a few traders, but the existence of such meetings will be unknown to non-Quanta QL users.

There seems to be a case for occasional general get-togethers, to which readers of both the Quanta magazine and *QL World* are invited. The organisation would almost certainly have to be handled by Quanta members, and there are quite a few of them who are both enthusiastic and experienced enough to do this. *QL World* now has a shorter lead-time for news to get into next issues, making it possible to advise readers of upcoming meetings in good time. Any comments?

Missing person

TF Services have tried, without success, to trace the owner of a QL, sent to them for repair in April 1992. The QL was repaired long ago, but letters advising of this, sent to the address given by the customer, have produced no reply. The customer's name is Mr Khuen (spelling uncertain), the address he gave was in Cheam, Surrey, and he said he was returning to Germany. If anyone reading this knows him, please contact TF Services (address given in INFORMATION below).

Bob McLaughlin has sent in some more information for users of the Panasonic KXP1123, KXP1124 and KXP1540 printers. Ribbons for these models can be obtained from the addresses given. Viking undertake to replace any ribbon that fails due to a manufacturing defect by six new ribbons! These ribbons are produced by Nukote International. Original Panasonic ribbons are obtainable from the same supplier and from several other suppliers who advertise regularly in computing magazines; they are likely to cost more than the "pattern" ribbons. One such supplier is listed below.

Complaints

Unusually, there has been a letters about problems concerning EEC. One of our regular correspondents over the years, James McGreehin, bought a monitor from EEC, and states that it did not work when received. He says that there was a short-circuit in the plug at the QL end of the connecting lead. He reports repairing the fault himself, and it is assumed the monitor has worked properly since. This complaint seems to be more about the fact that EEC did not acknowledge a letter sent claiming a 2% discount offered on goods ordered via the EEC answering machine, and reporting the faulty plug.

But a further angle concerns a matter that is perhaps less straightforward than he thinks. He ordered a DIY floppy disk drive kit from EEC but was informed it was not available. A complete 1 MB drive was offered instead, for £65, and the offer was accepted. When a drive arrived, it was a 2 MB type, but charged for at the offer price. There was a note with it, saying that it would

work OK as a 1 MB drive, but the instructions stated that DSDD (double-sided, double-density) disks might not format correctly in the 2 MB drive and HD disks should be used. Clearly, the thought of having to pay about twice the price for disks (HD instead of DD) does not appeal to James. Although he tried a DD disk and it appeared to be OK in the drive, there can be problems when using such drives with the QL; this is a general problem, not one associated only with EEC's drives.

James has a Sandy disk interface, and he had hoped to connect both dual 5.25-inch drives and the new 3.5-inch one to the same interface, but this interface does not support the connection of more than two drives. When he found this out, he contacted EEC and they offered to take the HD drive back. They were under no obligation to do so, as it is

the purchaser's job to determine whether or not what he orders is the correct unit for his purpose. A refund of £50 was made, the remaining £15 from the purchase price being retained to defray part of the cost of taking the drive out of stock and shipping it, then having to return

it to stock. James objects to not being refunded the full price, but this is standard commercial practice and the purchaser should not expect the supplier to absorb all the cost of a cancelled transaction. It would have been a different matter if EEC had given an assurance that the HD drive and the existing dual drives could be connected together to the particular Sandy interface and would work correctly then.

David Wyatt reports ordering and paying for a scanner from TK Computerware in July '92 and not receiving it as of 13th October. The unit was reportedly sent by Recorded

Delivery; in view of the number of times readers have told us they have not received goods stated by TK to have been sent Recorded Delivery, this statement needs to be treated with considerable reserve. Fortunately, David paid by credit card and should - eventually - get his money back. The address for Juergen Falkenburg, the producer of the QL Scanner, is given in the INFORMATION box, in case anyone wants to buy direct from him.

As no responses have been received from TK Computerware to several complaints referred to them many months ago, it has to be assumed that this supplier is not actively seeking QL business any more.

INFORMATION QL repairs:

TF Services
12 Bouverie Place
London W2 1RB.
Tel 071 724 9053
Fax 071 706 2379

QL Scanner:

Juergen Falkenburg
Thanweg 36
D-7359 Ersingen
Germany.

*Note that the **printer ribbon** prices shown below do not include shipping charges or VAT:*

Viking Direct
Bursom Industrial Park
Tollwell Road
Leicester LE4 1BR
Tel. Freephone 0800 424444
Fax Freefax 0800 622211

KXP1123/1124 ribbon £4.99
unbranded, £5.99 branded
KXP1540 ribbon £4.99
unbranded, £6.49 branded

CLP
Unit 7
Holland Way
Blandford
Dorset DT11 7TA
or
FREEPOST
Blandford
Dorset DT11 7BR.
Tel 0258 459544
Fax 0258 459565
KXP1123/1124 ribbon £3.80
unbranded, £7.29 branded
KXP1540 ribbon £5.19
unbranded, £8.17 branded.

document very thoroughly. Only then can you compile it without line numbers.

When everything worked well, I compiled the program again without line numbers and optimised for fast operation over the main loops. A document of 520 lines with lots of deliberate errors was completed in 69 seconds, having also inserted 51 sets of proportional toggles. Only one set was inserted when it should not have been, and about five sets were missed, but these were really on the border-line and the document suffered not in the least from their absence. There is room for improvement, especially with the hierarchy, which still gives me a few concerns but generally, the program is very useful and I am now so confident with it that if no errors are reported, I don't even check the proportional insertions.

The listing as such is unlikely to be exactly what you want. If you use different codes to perform the various

printer functions then either swap over to mine (!) or change the codes in the SElect algorithms and elsewhere accordingly. If you use a different wordprocessor altogether, you will need to understand something of the control-code structure that it uses to send things to the printer and the relationship between the wordprocessor itself and the printer-driver. *The Editor* is excellent in this respect since it so ridiculously simple. If you have any compiler other than *Turbo*, you might have to alter the program in other areas as a number of the keywords are taken from *Turbo Toolkit*.

First part

The Basic listing of my program, and a copy of my own printer driver were reproduced in the first part of this article, in *QL World* October 1992. However I am happy to supply any reader with a copy of the Basic listing, the compiled program, the printer driver plus a few

other associated odds and ends on disk for a small fee (£10, or £8 if sent a DD formatted diskette). Please apply to the author, 24 Eastcote Avenue, South Harrow, Middlesex, HA2 8AL with a suitable SAE and cheque. I would also be very encouraged to hear your own comments on the use of your printer, especially if it's of the Epson Protocol type. I cannot of course supply you with a copy of the XTRAS file. You must purchase this from Digital Precision (Toolkit), and the program will not run without it.

QL users are well known for the pride they take in their own systems over against bigger computer manufacturers. I do hope that this article will assist in enhancing the QL's character so as to produce work that is not only as good as but better than that of its rivals. I still believe that my basic 9 pin dot matrix printer, enhanced by our steadfast array of software houses plus a little creative thought of my own, produces documents

which have a nicer feel than quite a few laser printers that I have seen, operating under famous software. Happy typing!

Perfect Paragraphs & Shining Characters

Telephone: 0753-888866 (EEC) **W.N. Richardson & Co.**
Fax: 0753-887149

SINCLAIR QL PRICE LIST
DECEMBER '92

18-21, Miesbourne House, Chiltern Hill, Chalfont St Peter, SL9 9UE.
All Prices Include 17.5% VAT

QL Computers **MOST PRICES LOWER**

JS COMPLETE	6 Months Warranty, QL Computer, PSU, Psion Software (Quill, Archive, Abacus & Esel)	£120.00
	Free one years membership to QUANTA, the independent QL User Group with over 400 library programs, advice etc.	WITH JM ROM £ 95.00
PART EXCHANGE	A part exchange allowance of £30 is available against the above.	
JM BACKUP	Includes QL Computer and PSU only	£ 75.00
TINKERERS SPECIAL £49	(Please enquire for further details)	WITH JS ROM £ 85.00

Accessories * NOTE: EXTERNAL (SER2) 3 BUTTON MOUSE AND SOFTWARE WITH EXTRA FUNCTIONS. NOW HERMES COMPATIBLE *

PC KEYBOARD INTERFACE, INTERNAL FITTING, FOR FITTING 102 KEY KEYBOARD	£ 75.00
PC KEYBOARD, UK version, 102 key (AT)	£ 25.00
PC KEYBOARD INTERFACE AND KEYBOARD (INCLUDES FREE JOYSTICK OR PSU)	£ 90.00
CASE AND LEAD FOR EXTERNAL FITTING of Keyboard Interface	£ 17.00
JOYSTICK with QL lead (no interface required)	£ 10.00
* QL MOUSE, 3 Button, Software controlled, externally mounted, simply fits in SER2	£45 REMOTE MOUSE £ 55.00
MOUSE for QIMI interface, also OK for Atari ST and Amiga	£ 10.00

Disk Drives THE UNIVERSAL DRIVES ARE ALSO SUITABLE FOR ACORN BBC, ATARI ST, AMIGA, SPECTRUMS, IBM, AMSTRAD AND OTHER PC COMPATIBLES

Universal 3.5" 1Mb disk drive, cased with PSU and free QL lead	£ 70.00	2 Units for dual drive	£130.00
Universal 3.5" 2Mb disk drive, as above but 2Mb capacity	£ 90.00	2 Units for dual drive	£170.00
3.5" 1Mb Uncased disk drive (CITIZEN)			£ 25.00
3.5" 2Mb Uncased disk drive (CITIZEN)			£ 49.00
Lead for uncased disk drives, or other universal micros	£10.00	POWER SUPPLY FOR UNCASSED DRIVES	£ 6.00

Parallel Printers **NEW RANGE**

SAMSUNG SP093 80 Col, 300 cps, 50 NLQ, 3K Print Buffer, Centronics, Tractor Feed/Sheet Feed, Paper Park	£149.00
OLIVETTI COLOUR PRINTER, 24 pin, 200 cps, 50 NLQ, 40K Buffer, Epson LQ2550/IBM Compatible	£259.00
CENTRONICS PARALLEL Printer Interface for above printers	£ 27.00
QL LEAD FOR SERIAL PRINTERS	£ 10.00

Monitors GOLDSTAR 14" HIGH RES EGA Colour, 0.31 Dot Pitch, 85 COLUMN IN STYLISH GREY WITH NON GLARE SCREEN **£199**
BUY NOW WHILE STOCKS LAST TILT & SWIVEL BASE £5

Microdrive Cartridges and Spares

4 New Cartridges in a wallet	£ 10.00	8 prog carts for reformatting in 2 wallets	£ 15.00
Plastic Storage filing box including 20 new cartridges			£ 45.00
QL Psion Software V2.35 Includes Quill, Abacus, Archive, and Esel IN WALLET			£ 20.00
QL Psion Software Separate programs			£ 12.00
QL power Supply Unit	£ 10.00	Membrane (and instructions)	£ 9.00
TV or Network leads	£ 3.00	QL Top & Bottom Case	£ 5.00
IC's ZX 8301	£ 6.00	ZX 8302	£ 3.00
MC 68006	£ 12.00	8049 (IPC)	£ 3.00
		MC 1377	£ 3.00

(S.A.E. for other spares and repairs)

Payment terms:
CWO, Access, VISA et cetera
Cheques - allow 10 days

Delivery:
Carriage - £9
Postage - £3

Product subject to availability, E&OE.
Overseas - Add £10 for delivery

TEL: 0753 888866 FAX: 0753 887149

QUANTA

Independent QL/Thor Users Group

Worldwide Membership is by subscription only, and offers the following benefits:

- Monthly Newsletter - up to 40 pages
- Massive Software Library - mostly FREE!
- Free Helpline and Workshops
- Regional Sub-Groups. One near you?
- Advice on Software & Hardware problems
- Discounts from most major suppliers
- Subscription just £14 for UK members
- Overseas subscription £17

Barclaycard: Visa: Access: Mastercard

* Now in our NINTH successful year *

Further details from the Membership Secretary



Bill Newell
213 Manor Road
Benfleet
Essex. SS7 4JD
Tel (0268) 754 407



Eidersoft Ice rom

(windows, icons, pointers) front end, choice multitasking & mailmerge m'drive cartridge, plus 128k RAM extension in TECHNOLOGY RESEARCH Delta discdrive/printer interface. All manuals and documentation included. Complete package £60. Telephone 0732 355341

PC Conqueror original

£65 QL computer vers JM £60 super toolkit 2 EPROM cartridge £18 Centronics printer interface £18 Philips green Hi-res monitor BM7502 £70. Telephone 0233 626713 (Kent)

QL £50 green monitor

£30, real time digitizer £50, Twin disk drive £50, software: digital S.E. £25, editor S.E. £25, touch typist £6, assembler £12. Telephone 021449 8041

Unused Penman plotter

suitable for use with QL, PC, BBC and RM computers. Up to A2 size. Bargain at only £120. Tel: 0449 767130

QL 640k twin 3-1/2"

drives, mouse, minerva, cased separate keyboard £150. Tonto with monitor £100. Gram, painter, eye Q, CPM emulator, sprite generator, MDVS, games, books, mags. Tel: Jeff 0908 233008

QL Minerva Rom, Trump

Card 3-1/2" D/D, Miracle centronics (x2) Tandata modem, serial printer, 6 years QLWorld, Quanta mags, books, software, b/w television extra MDV. Sensible offers please. Tel: 0977 515428

QL JM Trump card 512k

TR. 128k interface books, programs £60 Tel: 081 998 8018

640k JS QL twin 3-1/2"

drives housed printer, monitor, schon keyboard, Tandata modem, eidersoft mouse. V2.35 Psion suite turbo Quill taskmaster spellbound speedscreen £220 ono. buyer collects. Tel: 0765 640331

QL unexpanded AH

version PSU Psion software manual etc. Any offers Miracle Systems QL modem £20. Phone: John 0689 855444

JS QL, Metal case, PC

keyboard and I/face, twin 3-1/2" d/drives and OPD m/drives, some software, new colour monitor £250 ono. Also Tandata modem 3, CQVI Digitiser and more. Tel: (Derby) 0332 677746

Dual disk drive £75

toolkit III £15 success £25 superforth £25 Digital C £20 Qload £10 QLiberator MDV £12. Better Basic £12. Gold card £200 HD/DD drive £100. Phone: Paul 0795 535928

Professional poolster REVOLUTIONARY NEW SYSTEM

COMBINES

Sound logic with A.I. to win £ hundreds £
Analysis of Bookies Factors for matches
Common sense improves chances of Mega-Win
UNIQUE ENTRY SYSTEM

Takes only a few minutes each week

Over 300 teams in data-bank.

Minor league teams automatically added.

Prior to PC release @ £99.00

To QL users offered at SPECIAL PRICE £49

THIS MONTH

MONEY BACK GUARANTEE - Make a profit in your first full season or your money back.

256K ROM minimum. Discs Only. State 3.5 or 5.25

**CHEQUE OR P.O. TO S.CANTON, OCEAN COMPUTER SERVICES, PO
BOX 68, NORTH DO. Manchester M8 6NJ
Tel. 061 740 9002**

Please send coupon to: Advertising Department Sinclair/QL World, The Blue Barn, Wootton, Woodstock, Oxon. OX7 1HA. Please place this Classified Advertisement in the next issue of Sinclair/QL world.

NAME

ADDRESS

DAYTIME TELEPHONE NO

I enclose a cheque/P.O. to the value of £17.65 (inc. VAT) for a lineage advertisement of no more than 30 words. Please make cheques payable to: Arcwind Ltd.

CLASSIFIED



Dilwyn Jones Computing

41 Bro Emrys, Tal-y-Bont,
Bangor, Gwynedd LL57 3YT U.K.

Tel: Bangor (0248) 354023

QL SOFTWARE

NB PLEASE NOTE THE MEANING OF CODES IN SQUARE BRACKETS BELOW:

[R] RAMDISK REQUIRED
[F] AVAILABLE ON FLOPPY DISK
[M] AVAILABLE ON MICRODRIVE
[128K/512K] MINIMUM MEMORY REQUIRED.
[PC] FOR IBM PC AND COMPATIBLES.
[TK2] TOOLKIT 2 REQUIRED.

CATALOGUE

THE CATALOGUE CONTAINS MORE DETAILS OF THE PROGRAMS. CALL FOR A COPY, OR ASK FOR A COPY WITH YOUR ORDER.

DEMONSTRATION VERSIONS

Demo versions are available for some of our programs. They cost £2.00 each and this cost is refundable against the full price of the full version, see lists below.

FLOPPY DISKS & LABELS

3.5" DSDD UNBRANDED DISKS £0.40
3.5" DSDD FLOPPY DISKS £0.70
MICRODRIVE CARTRIDGES £2.50
100 DISK LABELS £2.00
100 DISK LABELS ON PRINTER ROLL £2.50
100 ADDRESS LABELS ON ROLL £2.00
100 MICRODRIVE LABELS ON ROLL £2.00
Please add £0.50 postage per roll of labels (max postage £2.50 if only ordering labels. Add £2.50 per order for floppy disks. No extra postage on microdrive cartridges.

FILE TRANSFER

DISCOVER £20.00
[F 256k] Copy files QDOS to PC disks and vice versa.
MULTI DISCOVER £30.00
[F 256k] Enhanced versions, extra formats.
TEXTIDY
[F M 256k] Strip Quill files etc down to plain text.
OPD INTERCHANGE £15.00
ICL/OPD file transfer utility. Unsuitable Gold Card.
QL-PC FILESERVER £24.50
[F 128k] QL/PC software for serial cable link, access PC's drives, printer, transfer files, etc

LEISURE

SOLITAIRE £15.00
[F M 128k] A very addictive patience card game.
CRICKET SECRETARY £12.00
[F M 128k] For cricket buffs everywhere!
QUESTION MASTER £10.00
[F M 128k] For revision or entertainment.
QUIZ SETS each £5.00
FLEET TACTICAL COMMAND II QL £49.95
[F 512k] Naval strategy game, play on 2 networked QLs.
FLEET TACTICAL COMMAND II PC £69.95
[PC] Licensed PC version of this superb game.
COMBINED QL & PC VERSIONS £85.00
Package deal for both QL and PC versions together.
FTC2 DATA PRINT UTILITY £9.95
[F 128k] Print out data from FTCII described above.
THE FUGITIVE £9.95
[F 512k] Text adventure game.
COCKTAILS WAITER £10.00
[F 256k] Database of 100's alcoholic drinks recipes.
RECIPE SETS each £5.00

PROGRAMMING

S EDIT £20.00
[F 384k] DEMO VERSION £2.00
EASYPTR II £49.00
[F 256k] Pointer environment programming aid.
EASYPTR II BUDGET VERSION £35.00

DISA DISASSEMBLER

[F 256k] Interactive pointer driven disassembler. £29.00
BASIC REPORTER £10.00
[F M 128k] BASIC programming aid, list variables, etc.
BUDGET QLIVERATOR COMPILER £25.00
[F M 128k] BASIC compiler for unexpanded QL.
QLIVERATOR BASIC COMPILER £50.00
[F 256k] Excellent BASIC compiler for the QL.
QLOAD & QREF UTILITY £15.00
[F M 128k 128k] Fast loader and cross reference utility
MEGA TOOLKIT on disk £25.00
on EPROM cartridge and disk £40.00
Comprehensive toolkit of over 200 BASIC extensions

FILE HANDLING

LOCKSMITH £14.95
[M 128k] Microdrive backup program, byte by byte!
4MATTER £23.50
[F M 384k] Complete software backup system.
TOOLCHEST £14.95
[M F 256k] Create customised microdrive doctor
FILES 2 £12.00
[F M 128k] Simple to use popup file handling utility.
FILEMASTER £12.00
[F M 512k R] For bulk copying of files, & disk labelling.
THE GOPHER £12.00
[F M 128k] File finder, for disk, mdv, or hard disk.
[F M 128k] DEMO VERSION £2.00
WINBACK £25.00
[F 256k] Hard disk and ED disk backup utility
[F 256k] DEMO VERSION £2.00

DISPLAY SOFTWARE

BANTER £25.00
[F 512k] Banner printing program for Epson etc printers.
VISION MIXER 1 £10.00
[F 512k] Screen display system.
VISION MIXER PLUS £22.50
[F 384k] Enhanced, menu driven version
PICTUREMASTER £15.00
[F 256k] Screen drawing utility for use with V.Mixer, etc.
PICTUREMASTER PLUS £20.00
[F 384k] Enhanced version, extra features.
UPGRADE TO "PM PLUS" £5.00

PRINTER DUMPS

SIDEWINDER PLUS £24.95
[F M 512k] Screen and DTP page printer dump utility.
Suitable for 9 and 24 pin printers.

GRAPHICS

THE PAINTER V4.04 £25.00
[F 512k] 100% m/code pointer driven graphics program.
THE CLIPART £12.00
[F 128k] 3 disks packed full of screen pictures.
PICTORIAL INDEX FOR THE CLIPART £4.00
New printed index for the Clipart, with screen dumps
QRACAL £20.00
[F 512k] Machine code fractals program.
QRACAL SCREENS £2.00
[F 128k] Sample screen pictures made with QRacal.
IMAGE PROCESSOR 2 £15.00
[F 512k] Graphics and picture editing, etc utility.
[F 512k] DEMO VERSION £2.00
PD2 CLIPART £10.00
[F 128k] 2 disks of clipart screens.
SCREEN SNATCHER £10.00
[F M 128k] Grab screen displays from other programs.
TEXT 'N' GRAPHIX £20.00
[F 256k] Print text files containing screen dumps.
[F 256k] DEMO VERSION £2.00
TRANS24 £10.00
[F M 128k] Translate 9 pin graphics for 24 pin printer.
SCREEN COMPRESSION £10.00
[F M 128k] NEW! Save space by compressing pictures.

TEXT

QTPY 2 £29.95
[F 512k] Spelling checker, uses pointer environment.
BIBLE TEXT DISKS, PLAIN TEXT FORMAT £20.00
BIBLE TEXT DISKS, QUILL FILE FORMAT £20.00
[F 512k] Text of the King James Bible on disks.
SPELLBOUND £30.00
[F M 384k] Spelling checker, non-pointer environment.
SPELLBOUND SPECIAL EDITION £50.00
[F 512k] Improved version with larger dictionary, etc
UPGRADE TO SPECIAL EDITION £30.00
Please return master disk/cartridge when upgrading.
QUICK POSTERS £10.00
[F M 128k] Text poster maker for use with Star printers.

DATABASES

ADDRESS BOOK & LABEL PRINTER. £15.00
[F M R 384k] As used by DJC, database + label printer.
[F M R 384k] DEMO VERSION £2.00
DATA DESIGN £50.00
[F 512k] Programmable pointer environment database
FLASHBACK £25.00
[F M 256k] Fast database written in machine code.
FLASHBACK SPECIAL EDITION £40.00
[F 256k] Enhanced version of Flashback
QL GENEALOGIST SECOND EDITION £30.00
[F 384k] Family trees and family history program.
UPGRADE TO SECOND EDITION £12.00
Please return master disk with upgrade order.
BUDGET 128K GENEALOGIST £12.00
[F M 128k] Cut down version for unexpanded machines.
DISK INDEXER £12.00
[F M 256k] Make a database of contents of your disks.
DBEASY £15.00
[F 512k] Database front end for Archive etc.
DBPROGS £15.00
[F 512k] A collection of Archive utilities and text files

QL HARDWARE

MINI PROCESS CONTROLLER £59.95
Relay switched outputs, controlled via QL serial ports.
SOFTWARE TOOLKIT FOR MPC £9.95
[F 128k] Extensions to simplify control the MPC.
NETWORK PROVER £3.50
A small box which plugs between networked QLs.
QPOWER REGULATOR £24.95

OTHER QL SOFTWARE

QPAC1 £19.95
[F 512k] Pointer environment utilities.
[F 512k] Files maintenance menu, hotkeys, etc.
QTOP £29.50
[F 256k] Desktop system, files, job control, etc.
PRINTERMASTER £20.00
[F M 128k] Printer control utility.
HOME BUDGET £20.00
[F M 128k] Domestic bills and accounts program.
REMIND-ME £12.00
[F M 128k] Dates reminder program. Very easy to use.
REMIND-ME PLUS £20.00
[F M 128k] Enhanced version.
SCREEN ECONOMISER £10.00
[F M 128k] Screen saver blander.
SCREEN DAZZLER £15.00
[F M 384k] Like screen economiser, but instead of blanking out the screen activates moving objects or screen effects to save screen burn-ins, like some PC programs!
SLOWGOLD £5.00
[F 128k] Slowdown routine for Gold Card or any QL
TASKMASTER £25.00
[F M 384k] Task switching utility.
DISK LABELLER £10.00
[F 256k] Prints neat labels for your floppy disks
THE CAT £5.00
[F M 128k] List filenames in columns.
ROB ROY BARGAIN PACK £10.00
[F M 128k] Reviewed in Q L World August 1991

THE SMALL PRINT! POSTAGE AND PACKING CHARGES Software is sent post-free to UK addresses. To other countries, please add £1.00 per program for postage and packing (sent by airmail where possible). PRICES All prices are shown in UK Pounds Sterling. PAYMENT We can accept payment by cheque (in UK Pounds Sterling currency only, please) drawn on UK branch of a bank or building society, by Eurocheque with card number written on the back, Postal Order, or by these credit cards: VISA, ACCESS, MASTERCARD, EUROCARD or by CONNECT card. Please state the card type, number, expiry date, your address, and sign orders sent by post. We can also accept orders paid by credit card over the telephone. There is an answering machine for when I am unable to answer in person so that I can call you back later. Goods remain the property of DJC until paid for in full. PLEASE STATE IF YOU REQUIRE SOFTWARE ON 3.5 OR 5.25 INCH DISKS OR MICRODRIVE CARTRIDGE